

# ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

Τμήμα Μηχανικών Η/Υ, Τηλεπικοινωνιών & Δικτύων



## Διπλωματική Εργασία:

«Διδασκαλία προγραμματιστικών δομών σε πειραματικό  
διαδικτυακό περιβάλλον: Μια μελέτη περίπτωσης»

## Φοιτητής

Ορniθάς Αριστοτέλης

## Επιβλέποντες καθηγητές

Δασκαλοπούλου Ασπασία

Πολίτης Παναγιώτης

Βόλος, Μάρτιος 2009

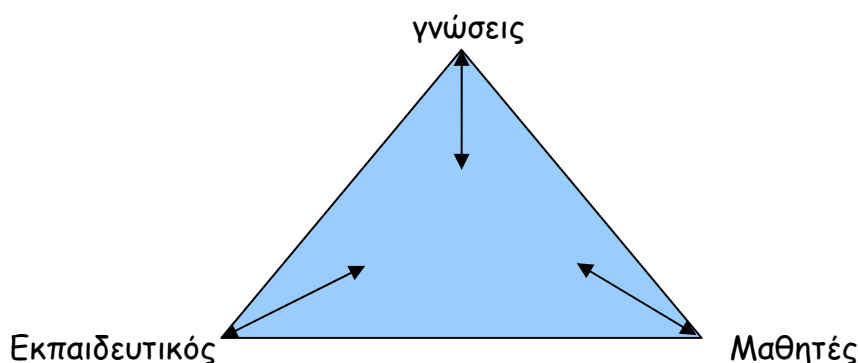
## Περιεχόμενα

1. Εισαγωγή.....	3
2. Το Μάθημα ΑΕΠΠ-Πρόγραμμα σπουδών.....	5
3. Το θεωρητικό μαθησιακό πλαίσιο.....	7
3.1. Εποικοδομητικά εκπαιδευτικά μαθησιακά περιβάλλοντα.....	7
4. Βασικές σχεδιαστικές επιλογές.....	9
4.1. Διαδικτυακό περιβάλλον.....	9
4.2. Αλληλεπιδραστικό περιβάλλον και ενεργητική μάθηση.....	9
4.3. Οπτικοποιημένο περιβάλλον.....	10
4.4. Περιβάλλον σύνταξης εκτέλεσης και ελέγχου αλγορίθμων.....	11
4.5. Java applets.....	12
5. Σημεία - θέματα επιλογής δομών και αλγορίθμων.....	14
5.1. Μεταβλητή.....	14
5.2. Δομές ελέγχου και η έννοια της επιλογής.....	15
5.2.1. Διδακτικά προβλήματα σχετικά με την έννοια της δομής ελέγχου.....	16
5.3. Έννοια της επανάληψης.....	16
5.3.1. Διδακτικά προβλήματα σχετικά με την έννοια της επανάληψης.....	17
6. Πειραματικό εκπαιδευτικό περιβάλλον.....	19
7. Σχέδια μαθήματος.....	29
7.1. Ταξινόμηση φυσαλίδας (bubblesort).....	29
7.2. Σειριακή Αναζήτηση.....	32
7.3. Δυαδική Αναζήτηση.....	34
8. Βιβλιογραφία.....	36

# 1. Εισαγωγή

Το θέμα της σχολικής μάθησης και της διδακτικής έχει γνωρίσει τα τελευταία χρόνια μια ιδιαίτερη θέση στην επιστημονική έρευνα και στην εκπαιδευτική κοινότητα. Η Διδακτική μελετά τις διαδικασίες μετάδοσης και πρόσκτησης των γνώσεων με απώτερο στόχο τη βελτίωση αυτών των διαδικασιών [Vergnaud, 1994]. Μελετά, συνεπώς, τις συνθήκες μέσα στις οποίες τα υποκείμενα μαθαίνουν ή δε μαθαίνουν, εστιάζοντας την προσοχή της στα ιδιαίτερα προβλήματα που προκύπτουν τόσο από το περιεχόμενο των γνώσεων όσο και των δεξιοτήτων που πρέπει να προσκτηθούν. [Κόμης, 2001].

Την διδακτική δηλαδή ενδιαφέρουν οι διαδικασίες που ευνοούν τον οικοδομητισμό των γνώσεων μέσα από καταστάσεις διδασκαλίας ατομικών ή συλλογικών. Μέχρι σήμερα δεν έχει καθιερωθεί μια θεωρία Γενικής διδακτικής και, για το λόγο αυτό γίνεται χρήση των εννοιών και των θεωρητικών πλαισίων που αναπτύσσονται στις διδακτικές άλλων μαθημάτων που έχουν διαμορφωθεί στη Διδακτική των Επιστημών. Η Διδακτική πολύ συχνά αναπαρίσταται από ένα τρίγωνο το οποίο συμβολίζει το σύστημα που συνδέει τις γνώσεις, το μαθητή και τον εκπαιδευτικό.



Οι τρεις κορυφές του τριγώνου αποτελούν η κάθε μια ξεχωριστό χώρο έρευνας από τη διδακτική και δεν πρέπει να εστιάσουμε εκεί αλλά στις σχέσεις που τις συνδέουν. Αυτό που ενδιαφέρει δηλαδή περισσότερο από διδακτική σκοπιά είναι οι διάφοροι τομείς που συνδέουν τις κορυφές [Κόμης, 2001]:

- Ο τομέας ανάπτυξης των περιεχομένων (*διδακτικός μετασχηματισμός, κοινωνικές πρακτικές αναφοράς*).
- Ο τομέας των στρατηγικών της οικοδόμησης των γνώσεων και της μάθησης (*αναπαραστάσεις, διδακτικά εμπόδια, επίλυση προβλήματος*).
- Ο τομέας της οικοδόμησης διδακτικών καταστάσεων (*διδακτικό σύμβολο, κατάσταση - πρόβλημα*).
- Ο τομέας των διδακτικών αλληλεπιδράσεων (*διδακτική βοήθεια κτλ.*) και των χρησιμοποιούμενων μέσων που παίζουν διαμεσολαβητικό ρόλο στις αλληλεπιδράσεις.

Κάτω από το πρίσμα αυτό, η ύπαρξη ενός τεχνολογικού μέσου, όπως ο υπολογιστής, δρα καταλυτικά κατά τη διδασκαλία της πληροφορικής ή κατά τη διδασκαλία με χρήση

υπολογιστή. Ο τομέας αυτός είναι ιδιαίτερα σημαντικός στη Διδακτική της Πληροφορικής, αφού κύριο ρόλο στη διδακτική πράξη διαδραματίζει η αλληλεπίδραση του μαθητή με το τεχνικό μέσο και οι παρεμβάσεις που παρέχει ο εκπαιδευτικός.

## **2. Το Μάθημα ΑΕΠΠ-Πρόγραμμα σπουδών**

Το μάθημα «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον» είναι το βασικό μάθημα του κύκλου «Υπηρεσιών και Πληροφορικής» του Ενιαίου Λυκείου και ο γενικός του σκοπός περιγράφεται παρακάτω.

Ο γενικός σκοπός του μαθήματος είναι να αναπτύξουν οι μαθητές αναλυτική και συνθετική σκέψη, να αποκτήσουν ικανότητες μεθοδολογικού χαρακτήρα και να μπορούν να επιλύουν απλά προβλήματα σε προγραμματιστικό περιβάλλον. [Πηγή: Αναλυτικό Πρόγραμμα Σπουδών] Η διδακτέα ύλη του μαθήματος ΑΕΠΠ οργανώνεται σε δυο κύρια μέρη. Το πρώτο είναι το «θεωρητικό» περιλαμβάνει τα κεφάλαια 1-5 και δίνει έμφαση στην ανάπτυξη αλγοριθμικής προσέγγισης των προβλημάτων. Το δεύτερο μέρος αφιερώνεται στην υλοποίηση προγραμμάτων σε προγραμματιστικό περιβάλλον. Τα δυο αυτά μέρη είναι όμως αλληλένδετα και δεν διδάσκονται σειριακά αλλά παράλληλα.

Η προσέγγιση των εννοιών και η καλλιέργεια δεξιοτήτων που απαιτούνται για την υλοποίηση του γενικού σκοπού του μαθήματος «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον» ταξινομούνται σε τέσσερις άξονες [Πηγή: Αναλυτικό Πρόγραμμα Σπουδών] :

### **Ανάλυση του προβλήματος**

- Καθορισμός και κατανόηση του προβλήματος
- Μεθοδολογίες ανάλυσης
- Καθορισμός απαιτήσεων
- Κύκλος ζωής λογισμικού

### **Σχεδίαση του αλγόριθμου**

- Προδιαγραφές σχεδιασμού
- Μεθοδολογίες σχεδιασμού
- Σύγχρονες τεχνικές σχεδίασης προγραμμάτων
- Αλγοριθμική γλώσσα
- Ανάπτυξη αλγόριθμου
- Έλεγχος αλγόριθμου
- Σχεδιασμός περιβάλλοντος διεπαφής

### **Υλοποίηση σε προγραμματιστικό περιβάλλον**

- Δομημένος προγραμματισμός
- Αντικειμενοστραφής προγραμματισμός
- Δομικά στοιχεία προγραμματισμού
- Σύγχρονα προγραμματιστικά εργαλεία
- Εκσφαλμάτωση προγράμματος
- Επικοινωνία με άλλες εφαρμογές

### **Τεκμηρίωση -Αξιολόγηση**

- Τεκμηρίωση του προγράμματος
- Αξιολόγηση της απόδοσης του προγράμματος
- Περιγραφή πιθανών εναλλακτικών λύσεων
- Δυνατότητες επέκτασης και όρια χρήσης

Κατά την **«ανάλυση και σχεδίαση του αλγορίθμου»** Οι μαθητές κατανοούν το πρόβλημα, το αναλύουν, προσεγγίζουν με αυστηρότητα την έννοια του αλγόριθμου και περιγράφουν την αλγοριθμική διαδικασία επίλυσής του.

Στη φάση **«Υλοποίηση σε προγραμματιστικό περιβάλλον»** Οι μαθητές μαθαίνουν να χρησιμοποιούν προγραμματιστικά εργαλεία, να εφαρμόζουν προγραμματιστικές τεχνικές, να γράφουν το πρόγραμμα, να το εκτελούν, να το διορθώνουν και να το βελτιώνουν.

Και τέλος κατά την **«Τεκμηρίωση - Αξιολόγηση»** Οι μαθητές τεκμηριώνουν την εργασία τους και αξιολογούν την ποιότητά της.

### 3. Το Θεωρητικό μαθησιακό πλαίσιο

#### 3.1. Εποικοδομητικά εκπαιδευτικά μαθησιακά περιβάλλοντα

Η διδασκαλία που βασίζεται στη θεωρία του εποικοδομητισμού θέλει από τους εκπαιδευόμενους να είναι οι ίδιοι οι δημιουργοί των γνώσεων τους και στοχασμού πάνω στα προβλήματα που προκύπτουν κατά την μαθησιακή διαδικασία. Επομένως τα υποκείμενα δημιουργούν τις δικές τους εικόνες και αναπαραστάσεις των καινούριων πληροφοριών που τους προσφέρονται αναθεωρώντας και επανεξετάζοντας τις εκτιμήσεις τους. Την διδασκαλία βασισμένη στον εποικοδομητισμό εξυπηρετεί σε μεγάλο βαθμό η αλληλεπιδραστική σχέση του εκπαιδευόμενου με το εκπαιδευτικό λογισμικό. Ο Hundhausen (2002), μετά από την ανάλυση των αποτελεσμάτων 24 πειραματικών διδασκαλιών, κατέληξε στο συμπέρασμα ότι *"όσο πιο ενεργά συμμετέχουν οι μαθητές σε δραστηριότητες με το εκπαιδευτικό λογισμικό, τόσο καλύτερα αποδίδουν αργότερα όταν καλούνται να υλοποιήσουν τον αλγόριθμο που διδάχθηκαν"*.

Κατά την αλληλεπίδραση του εκπαιδευόμενου με το εκπαιδευτικό λογισμικό ο εκπαιδευόμενος είναι ενεργό μέρος της μαθησιακής διαδικασίας αναλαμβάνοντας ο ίδιος πρωτοβουλίες και οικοδομώντας τις γνώσεις που προσφέρονται μέσα από ένα διαδραστικό περιβάλλον καθιστώντας τον, αντί για παθητικό δεκτή της πληροφορίας, δημιουργό των γνώσεων και των αναπαραστάσεων. Αυτή είναι άλλωστε και η θεμελιώδης αρχή του εποικοδομητισμού. Η γνώση λοιπόν κατασκευάζεται από το ίδιο το υποκείμενο κατά την αλληλεπίδρασή του με το μέσο διδασκαλίας. Τα υποκείμενα μέσω της αλληλεπιδραστικής σχέσης με την εφαρμογή έχουν τη δυνατότητα να προβληματιστούν και να σκεφτούν οι ίδιοι λύσεις σε διάφορες απορίες και προβληματισμούς που προκύπτουν κατά τη διάρκεια της εκπαίδευσης.

Στο πλαίσιο της εποικοδομητικής προσέγγισης έχουν αναπτυχθεί διάφορα διδακτικά μοντέλα που περιλαμβάνουν τις εξής φάσεις (Κόκκοτας, 1998):

**Φάση του προσανατολισμού.** Πρόκειται για την εισαγωγική φάση πρόκλησης του ενδιαφέροντος των εκπαιδευόμενων, με τις κατάλληλες ερωτήσεις και δραστηριότητες.

**Φάση της ανάδειξης των ιδεών των εκπαιδευόμενων.** Πρόκειται για την προφορική ή γραπτή έκφραση των ιδεών των εκπαιδευόμενων. Η φάση είναι από τις σημαντικότερες της διδακτικής διαδικασίας, όχι μόνο επειδή ο εκπαιδευόμενος εξωτερικεύει τις ιδέες του, αλλά γιατί και ο εκπαιδευτής γίνεται γνώστης της σκέψης των εκπαιδευόμενων και έτσι μπορεί να επιλέξει τις κατάλληλες διδακτικές στρατηγικές (Κόκκοτας, 1998).

**Φάση της αναδόμησης των ιδεών των εκπαιδευόμενων.** Πρόκειται για τη φάση που οι εκπαιδευόμενοι προχωρούν σε έλεγχο των ιδεών τους με σκοπό να οδηγηθούν στην υιοθέτηση απόψεων συμβατών με το επιστημονικό πρότυπο. Οι εκπαιδευόμενοι ενθαρρύνονται να πειραματιστούν και να διερευνήσουν την λειτουργικότητα και αποτελεσματικότητα των απόψεων που ήδη έχουν διατυπώσει είτε προσωπικά είτε ως ομάδα. Αυτή η έμπρακτη διερεύνηση των εκπαιδευόμενων θα τους οδηγήσει είτε

στην επιβεβαίωση των ιδεών τους είτε στη διαπίστωση πως αυτές οι ιδέες αντιφάσκουν με το πειραματικό αποτέλεσμα. Στην πρώτη περίπτωση οι ιδέες αυτές ισχυροποιούνται αλλά και διευρύνονται (Κόκκοτας, 1996). Στη δεύτερη περίπτωση ο εκπαιδευόμενος οδηγείται σε γνωστική σύγκρουση.

**Φάση της εφαρμογής των ιδεών.** Πρόκειται για την εφαρμογή της νέας γνώσης σε πραγματικές καταστάσεις και στις δραστηριότητες που πραγματοποιήθηκαν σε προηγούμενες φάσεις και οδήγησαν τους εκπαιδευόμενους σε γνωστική σύγκρουση.

**Φάση της ανασκόπησης.** Πρόκειται για το μεταγνωστικό στάδιο κατά το οποίο οι εκπαιδευόμενοι αναστοχάζονται πάνω στη σημασία της νέας γνώσης που ανακάλυψαν, μέσα από μια διαδικασία σύγκρισης των προηγούμενων ιδεών με αυτές που υιοθετήθηκαν στο τέλος της διδασκαλίας.



## **4. Βασικές σχεδιαστικές επιλογές**

### **4.1. Διαδικτυακό περιβάλλον**

Το διαδίκτυο και η ευρεία αποδοχή του World Wide Web (WWW) ως καθολικού μέσου για τη διάδοση και την επεξεργασία πληροφοριών που είναι προσβάσιμες με τη μορφή πολυμέσων και σε εξαιρετικά μεγάλες ταχύτητες επικοινωνίας, έχει ανοίξει νέες προοπτικές στην εκπαίδευση με τη προσπάθεια να εκμεταλλευτεί πλήρως τις βασικές μας "αισθήσεις" για μάθηση. Το διαδίκτυο λοιπόν μας προσφέρει απεικόνιση των τρισδιάστατων αντικειμένων, μορφοποιημένο κείμενο, έννοιες που απαιτούν για την καταλληλότερη παρουσίασή τους στατικές ή κινούμενες εικόνες, αλληλεπιδραστικά στοιχεία, ήχους ή βίντεο. Επιλέχθηκε λοιπόν η HTML μορφή για τη συγγραφή των δραστηριοτήτων, επειδή είναι η πιο διαδεδομένη μορφή που υποστηρίζει τα ανωτέρω χαρακτηριστικά. Εκτός όμως από αυτά, λόγω της ανοιχτής μορφής της η HTML προσφέρει και άλλους τρόπους αναπαράστασης: τρισδιάστατα αντικείμενα VRML, java applets, κώδικα Javascript (DHTML) και πολλά ActiveX controls όπως είναι το πολύ γνωστό Macromedia Flash Player. Έτσι με την HTML μορφή προσδίδουμε στις δραστηριότητες όλες τις αλληλεπιδραστικές και πολυμεσικές δυνατότητες του Διαδικτύου.

### **4.2. Αλληλεπιδραστικό περιβάλλον και ενεργητική μάθηση**

Ο όρος "αλληλεπίδραση" χρησιμοποιείται συχνά για να περιγράψει μια διαδικασία κατά την οποία το άτομο έρχεται σε επαφή με το περιβάλλον του. Η φύση και η έκταση της αλληλεπίδρασης που θα μπορούσε να χαρακτηριστεί ως κατάλληλη για οποιοδήποτε μαθησιακό περιβάλλον ποικίλει σύμφωνα με τη διδακτική φιλοσοφία, τη φύση του γνωστικού αντικειμένου διδασκαλίας, την ωριμότητα των φοιτητών/εκπαιδευόμενων και του μέσου που χρησιμοποιείται στο μάθημα (Liaw & Huang, 2000, Trentin, 2000, Yacci, 2000). Η αλληλεπίδραση προσεγγίζεται ως μια από τις σημαντικότερες διαδικασίες και χαρακτηριστικό στην εκπαιδευτική διαδικασία, ιδιαίτερα όταν βασίζεται στις αρχές της ενεργητικής μάθησης και της οικοδόμησης της γνώσης, Στο πλαίσιο αυτών των αρχών οι εκπαιδευόμενοι δεν εκλαμβάνονται ως παθητικοί δέκτες αλλά ως αυτόνομα και υπεύθυνα άτομα τα οποία συμμετέχουν ενεργά στη διαδικασία της μάθησης, υποβοηθούμενοι από την αλληλεπίδραση τους με το εκπαιδευτικό λογισμικό.

Η έννοια της ενεργητικής μαθησιακής αυτονομίας προϋποθέτει, αλλά και συνεπάγεται, ότι ο εκπαιδευόμενος θα αναπτύξει ένα ιδιαίτερο είδος ψυχολογικής σχέσης με τη διαδικασία και το περιεχόμενο της μάθησης καθώς και με τον τρόπο με τον οποίο μεταφέρει ότι έχει μάθει σε άλλες κοινωνικές καταστάσεις (Little, 1991). Η παρέμβαση του εκπαιδευτικού είναι η υποστήριξη των μαθησιακών επιδιώξεων των εκπαιδευόμενων με ταυτόχρονη σταδιακή απόσυρση του ελέγχου και της υποστήριξης από το εκπαιδευτικό λογισμικό. Δημιουργείται, με άλλα λόγια, ένα πλαίσιο στήριξης (scaffolding) το οποίο μπορεί να οδηγήσει σταθερά τον εκπαιδευόμενο στην αυτόνομη και αυτορρυθμιζόμενη μάθηση μέσω της

αλληλεπίδρασης με το μέσο. έχει υποστηριχθεί επίσης ότι η μάθηση γίνεται περισσότερο αποτελεσματική όταν ο εκπαιδευόμενος έχει τον έλεγχο της μαθησιακής διαδικασίας, συμμετέχει ενεργά στην οικοδόμηση της γνώσης και έχει τη δυνατότητα διαλεκτικής επικοινωνίας σε ένα συνεργατικό περιβάλλον μάθησης (Resta, 1998, Ryan & Woodward, 1998, Seffah & Bouchard, 1998, Carvin, 1997, Goetter & Kazamek, 1990).

Παράλληλα, η γραφιστική προσέγγιση στην αλληλεπίδραση, όσο αφορά στην εργονομική διάταξη και στη γραφιστική απεικόνιση και μεταφορά, παίζει σημαντικό ρόλο στα μαθησιακά κίνητρα και κατά συνέπεια στην έκβαση του μαθησιακού αποτελέσματος (Flottemech, 2000, Lohr, 2000, Lee & Bolding, 1999, Rieber). Είναι όμως γνωστό ότι οι σπουδαστές ελκύονται συχνά από την καινοτομία των αλληλεπιδραστικών πολυμέσων, τουλάχιστον αρχικά, αλλά εάν τα κίνητρα δεν προεκτείνονται πέρα από το επίπεδο της καινοτομίας, η ενεργός εμπλοκή των φοιτητών στα μαθησιακά δρώμενα δεν παραμένει για πολύ. Η διάταξη της πληροφορίας στην οθόνη και ο τρόπος παρουσίασής της επηρεάζει επίσης σημαντικά την κατεύθυνση, τη διάρκεια και το βαθμό προσοχής του φοιτητή.

Στο πλαίσιο αυτό, το χρώμα, η ταχύτητα, η κίνηση και ο ήχος, πέρα από τα γλωσσικά συστατικά στοιχεία μιας οθόνης, μπορούν να επηρεάσουν την αλληλεπιδραστική διαδικασία. Το χρώμα μπορεί να συνεισφέρει στην επισήμανση των σημαντικών πληροφοριών, να βοηθήσει την οργάνωση του περιεχομένου του μαθήματος, να επιτρέψει το λογικό έλεγχο των επιλογών των φοιτητών, να προωθήσει την αλληλεπίδραση μεταξύ του φοιτητή και του περιεχομένου του μαθήματος και να διευκολύνει την πλοήγηση στο μάθημα (Hannafin & Hooper, 1989).

Οι περισσότεροι ερευνητές φαίνονται να συμφωνούν ότι η γραφιστική απεικόνιση δημιουργεί κίνητρα μάθησης, αν και η έρευνα δείχνει ότι το παθητικό αποτέλεσμα των γραφιστικών απεικονίσεων ποικίλει ανάλογα με την ηλικία, την ευφυΐα και το μορφωτικό επίπεδο του χρήστη (Hartley, 1995). Το ίδιο θα μπορούσαμε να ισχυριστούμε και για τη χρήση άλλων τύπων (π.χ. ήχος, κινούμενο σχέδιο) και μέσων αναπαράστασης της πληροφορίας (Μακράκης, 2000).

### **4.3. Οπτικοποιημένο περιβάλλον**

Με τον όρο *οπτικοποίηση (visualization)* αλγορίθμου περιγράφεται μια διαδραστική οπτικοποιημένη παρουσίαση της λογικής που κρύβεται πίσω από τον αλγόριθμο, βασισμένη σε μια σειρά εικόνων και αναπαραστάσεων που εστιάζουν στα βασικά χαρακτηριστικά της συμπεριφοράς του (Hundhausen & Brown, 2007). Μια απλή αναπαράσταση ενός αλγορίθμου με χρήση πολυμέσων δεν είναι αρκετή. Πρόσφατες έρευνες έχουν δείξει ότι τα λογισμικά προσομοίωσης αλγορίθμων θα πρέπει να προωθούν τη διερευνητική μάθηση και να ευνοούν την ενεργητική συμμετοχή και όχι την παθητική συμμόρφωση των μαθητών (Hundhausen et.al., 2002; Hundhausen & Brown, 2006) Ο μαθητής δεν αρκεί απλά να παρακολουθεί την παρουσίαση ως παθητικός θεατής, αλλά θα πρέπει να πειραματιστεί με τον αλγόριθμο, έτσι ώστε να κατανοήσει πλήρως και σε βάθος τη λογική και τη λειτουργικότητα του. Οι

ερευνητικές μελέτες σχετικά με την αποτελεσματικότητα των συστημάτων οπτικοποίησης-προσομοίωσης αλγορίθμων έδειξαν ότι τα μαθησιακά οφέλη είναι περιορισμένα όταν ο μαθητής παρακολουθεί απλά μια οπτικοποίηση χωρίς να εμπλέκεται σε γνωστικές διαδικασίες χειρισμού και αλληλεπίδρασης με τον αλγόριθμο (Byrne et al., 1999; Hundhausen & Brown, 2006; Stasko, 1993; 1997). Υπάρχουν πολλοί λόγοι που συνηγορούν στη χρήση των προσομοιώσεων-οπτικοποιήσεων στα εισαγωγικά μαθήματα του προγραμματισμού. Οι προσομοιώσεις αλγορίθμων αποτελούν δυναμικά μαθησιακά περιβάλλοντα τα οποία (Τζιμογιάννης κ.α., 2006):

- επιτρέπουν την οπτικοποίηση της λειτουργίας ενός αλγορίθμου, υποστηρίζοντας τη διάλεξη του εκπαιδευτικού και την εργαστηριακή εξάσκηση των μαθητών
- επιτρέπουν στους μαθητές να απομονώσουν τις μεταβλητές και το ρόλο τους στο πρόγραμμα, με στόχο την κατανόηση πιο σύνθετων υπολογιστικών δομών και διαδικασιών
- βοηθούν τους μαθητές να εμβαθύνουν στη λογική του προγραμματισμού και να κατανοήσουν δύσκολες υπολογιστικές έννοιες και διαδικασίες
- ενεργοποιούν το ενδιαφέρον των μαθητών, παρέχοντας δυνατότητες να εκφράσουν τις δικές τους αναπαραστάσεις για αλγορίθμους και διαδικασίες
- διευκολύνουν την ενεργητική μάθηση μέσα από διαδικασίες παράθεσης υποθέσεων, μεταβολής των τιμών εισόδου και άμεσου ελέγχου των αποτελεσμάτων στην οθόνη
- βοηθούν τους μαθητές να μάθουν μέσα από διερευνητικές δραστηριότητες.

#### ***4.4. Περιβάλλον σύνταξης εκτέλεσης και ελέγχου αλγορίθμων***

Στη διδασκαλία του προγραμματισμού η πρακτική εφαρμογή των γνώσεων της θεωρίας είναι κάτι παραπάνω από απαραίτητη λόγω της φύσης του μαθήματος. Έτσι η διαδικασία της επίλυσης ασκήσεων προγραμματισμού σε ένα συμβατό με τη θεωρία προγραμματιστικό περιβάλλον είναι αναπόσπαστο κομμάτι μιας ολοκληρωμένης διδασκαλίας. Μέσα από την επίλυση προβλημάτων σε ένα προγραμματιστικό περιβάλλον οι εκπαιδευόμενοι έχουν τη δυνατότητα να εξερευνήσουν την όλη διαδικασία εκτέλεσης του προγράμματος, που σε άλλη περίπτωση μένει κρυφή από αυτούς. Με την εξερεύνηση της διαδικασίας του τρεξίματος ενός προγράμματος και τον ενεργό πειραματισμό προάγεται η δημιουργικότητα των μαθητών καθώς και η εποικοδομητική μάθηση.

Οι μαθητές έρχονται αντιμέτωποι με σφάλματα (bugs), άμεσα παρατηρήσιμα στην οθόνη του υπολογιστή, που προκαλούν οι παρανοήσεις τους και πιθανώς οι λανθασμένες νοητικές εικόνες δημιουργώντας έτσι ανάδραση ως συνέπεια των λαθών τους. Μέσα από τον προβληματισμό και την διόρθωση των λογικών και συντακτικών λαθών ενισχύεται η μαθησιακή αποτελεσματικότητα και αποτελεί στοιχείο αυτορρυθμιζόμενης μάθησης. Με τον όρο αυτορρύθμιση εννοούμε την διαδικασία στην οποία οι μαθητές ενεργοποιούν και διατηρούν γνωστικές

λειτουργίες, συμπεριφορές και συναισθήματα, τα οποία προσανατολίζονται στην επίτευξη των στόχων τους (Schunk & Zimmerman, 1994).

Εν κατακλείδι η χρήση κατάλληλου προγραμματιστικού περιβάλλοντος έχει μεγάλη παιδαγωγική αξία γιατί επιτρέπει τη σύνταξη, τον έλεγχο και τη διόρθωση του προγράμματος από τους ίδιους τους μαθητές. Με άλλα λόγια, το προγραμματιστικό περιβάλλον παρέχει σε κάθε μαθητή δυνατότητες εξάσκησης, ενεργού πειραματισμού, ελέγχου των νοητικών μοντέλων για τα υπολογιστικά αντικείμενα και τις προτεινόμενες λύσεις, συνεργασίας και αλληλεπίδρασης με τους συμμαθητές του.

#### **4.5. Java applets**

Όπως αναφέρθηκε παραπάνω η οπτικοποίηση της γνώσης κατέχει εξέχοντα ρόλο στη μαθησιακή διαδικασία. Μια απλή οπτικοποίηση της γνώσης μέσω πολυμέσων, αν και επιθυμητή, δεν είναι αρκετή μιας και αναφέρθηκε η σπουδαιότητα της δυναμικής οπτικοποίησης της γνώσης και η μεταβολή του υποκειμένου από παθητικό θεατή σε ενεργό μέλος της οικοδόμησης της γνώσης του. Σε αυτό το σημείο οι μικροεφαρμογές (applets) της JAVA έρχονται να αναβαθμίσουν τις παραδοσιακές τεχνικές εκπαίδευσης και να εισάγουν ένα καινούριο μαθησιακό μοντέλο με έμφαση στον εκπαιδευόμενο εξυπηρετώντας τις αρχές του εποικοδομητισμού και της δυναμικής προσομοίωσης της γνώσης.

Υπάρχουν τουλάχιστον τέσσερις τρόποι με τους οποίους τα applets μπορούν να χρησιμοποιηθούν στην εκπαίδευση (Σ. Πανέτσος, Ι. Κασιρίης):

- **Ενημερωτικά Applets.** Αυτά τα applets είναι παρόμοια με τα αρχεία βοήθειας στα προγράμματα των WINDOWS.
- **Applets επεξηγηματικά εννοιών.** Τέτοια applets επεξηγούν τις έννοιες που συνθέτουν το θέμα, για παράδειγμα, η έννοια των σειρών Fourier στην φασματική ανάλυση.
- **Υπολογιστικά Applets.** Αυτά τα applets μπορούν να χρησιμεύσουν ως παραδείγματα εκμάθησης των εννοιών καθώς επίσης και για την επεξήγηση φαινόμενων. Μπορεί να έχουν την ικανότητα διαδραστικής απεικόνισης των αποτελεσμάτων με ενσωματωμένο GUI και να διευκολύνουν τον πειραματισμό με το χειρισμό των διάφορων παραμέτρων που μπορούν μεταβληθούν.
- **Applets αξιολόγησης.** Για να αξιολογηθεί η εκπαίδευση των σπουδαστών, μπορεί να σχεδιαστούν applets διαγωνισμάτων με ερωτήσεις πολλαπλής επιλογής.

Με βάση τις παραπάνω λειτουργικότητες που επιτελούν τα Java applets μπορούν να περιγράφουν έννοιες δυσνόητες προς τους μαθητές και να μεταδοθούν πληροφορίες που με την παραδοσιακή διδασκαλία είναι δύσκολο να μεταδοθούν. Τέτοια παραδείγματα είναι κάποιες βασικές έννοιες που περιλαμβάνονται σε κάποιους αλγόριθμους όπως για παράδειγμα η αντιμετάθεση δυο στοιχείων ενός

πίνακα στον αλγόριθμο της φυσαλίδας, μια έννοια δύσκολη προς τους αρχάριους στον προγραμματισμό να την καταλάβουν εκτός και αν την δουν οπτικοποιημένη διαμορφώνοντας ή επαναπροσδιορίζοντας νοητικές εικόνες που του δημιουργήθηκαν στη θεωρία.

Κάποιες κατηγορίες πληροφοριών που αλλιώς είναι δύσκολο να μεταδοθούν είναι:

**Δυναμικές πληροφορίες**, όπως η προσομοίωση μιας χημικής αντίδρασης ή ο ρυθμός σύγκλισης μιας αριθμητικής μεθόδου ολοκλήρωσης ή κίνησης ενός εκκρεμούς.

**Πληροφορίες πολυμέσων**, όπως οποιαδήποτε θέματα που περιλαμβάνουν τη μελέτη εννοιών με γραφική παράσταση, κίνηση ή ήχο.

**Διαδραστικές πληροφορίες**, όπως οποιοδήποτε θέμα που περιλαμβάνει την αλλαγή των παραμέτρων του συστήματος με σκοπό την κατανόηση.

Ένας ακόμη παράγοντας που συνηγορεί στη χρήση τέτοιων εφαρμογών είναι το κόστος καθώς οι εμπορικές εφαρμογές και οι γλώσσες προγραμματισμού δεν είναι εύκολα προσιτές στους μαθητές σε αντίθεση με τους browsers που δεν έχουν κόστος και μπορούν να εκτελούν τέτοιου είδους εφαρμογές. Τέλος πέρα από τη χρησιμότητα των applet για τη όλα τα παραπάνω μπορούν να χρησιμοποιηθούν και ως εργαλεία αξιολόγησης των εκπαιδευόμενων καθώς οι παραδοσιακοί τρόποι ελέγχου και αξιολόγησης της αποκτηθείσης γνώσης μπορούν να αντικαταστούν από on-line tests πολλαπλής επιλογής βασισμένα σε applets.

## 5. Σημεία – Θέματα επιλογής δομών και αλγορίθμων

Η απόφαση σχετικά με τον προσδιορισμό των κεφαλαίων της ύλης βασίστηκε στο περιεχόμενο, στην οργάνωση βιβλίων γλωσσών προγραμματισμού και στις μαθησιακές δυσκολίες που συναντούν οι μαθητές. Έτσι, τα κεφάλαια της εφαρμογής μας αφορούν σε αλγορίθμους αναζήτησης και ταξινόμησης. Συγκεκριμένα υπάρχουν οι αλγόριθμοι σειριακής και δυαδικής αναζήτησης και ο αλγόριθμος ταξινόμησης φουσαλίδας (bubblesort). Η επιλογή αυτών των κεφαλαίων βασίστηκε όχι μόνο στα περιεχόμενα των σχετικών βιβλίων, αλλά και στη λογική που διέπει τις γλώσσες προγραμματισμού. Για παράδειγμα, η εκμάθηση ενός αλγορίθμου ταξινόμησης προϋποθέτει τη γνώση των δομών επιλογής και επανάληψης. Με την επιλογή αυτών των δομών και αλγορίθμων σαν αντικείμενο μελέτης γίνεται μια συνολική ανασκόπηση των πιο βασικών μαθησιακών προβλημάτων που αντιμετωπίζουν οι μαθητές. Αναλυτικότερα στην επαφή τους οι εκπαιδευόμενοι με αυτούς τους αλγορίθμους έρχονται αντιμέτωποι με τις έννοιες και τα διδακτικά τους προβλήματα που έχουν διδαχτεί πιο πριν και είχαν τυχόν μαθησιακές δυσκολίες. Συγκεκριμένα αντιμετωπίζουν :

- Την έννοια της μεταβλητής
- δομές ελέγχου και την έννοια της επιλογής
- έννοια της επανάληψης

Κάθε μια από τις οποίες έχει τα δικά της διδακτικά προβλήματα

### 5.1. Μεταβλητή

Η πρόσκτηση της έννοιας της μεταβλητής στον προγραμματισμό είναι ιδιαίτερα σημαντική και παράλληλα αρκετά δύσκολη. Παρά τις διαφοροποιήσεις, η πλειονότητα των γλωσσών προγραμματισμού βασίζεται στην ίδια θεμελιώδη τεχνική, αυτή της διαχείρισης τιμών που περιέχονται σε μεταβλητές [Delannoy, 1996]. Η δραστηριότητα του προγραμματισμού απαιτεί σε μεγάλο βαθμό τη γνώση χρήσης συμβόλων και αναπαραστάσεων για τη διαχείριση και την αποθήκευση δεδομένων. Το ζήτημα της αποθήκευσης των δεδομένων με τη μορφή μεταβλητών, οι οποίες γίνονται αντιληπτές με τη χρήση συμβολικών κωδικών, συνιστά εντούτοις ένα δύσκολο πρόβλημα στον προγραμματισμό [Dufoyer, 1988].

Το πιο σημαντικό μαθησιακό πρόβλημα προέρχεται από το γεγονός ότι η έννοια της μεταβλητής στον προγραμματισμό δεν ταυτίζεται με αυτήν των μαθηματικών. Η κατανόηση της έννοιας της μεταβλητής στον προγραμματισμό μπορεί να βασιστεί στην προηγούμενη γνώση που έχουν οι εκπαιδευόμενοι από τα μαθηματικά αλλά αυτό είναι άλλες φορές βοήθεια και άλλες φορές διατακτικό εμπόδιο. Ο στατικός όμως χαρακτήρας που έχει η μεταβλητή στα μαθηματικά αποτελεί συνήθως διδακτικό εμπόδιο στην αναπαράσταση της δυναμικής τροποποίησης της τιμής της μεταβλητής κατά την εκτέλεση του προγράμματος [Rogalski & Vergnaud, 1987].

Η εκχώρηση ή ανάθεση τιμής μιας μεταβλητής συνιστά βασικό διδακτικό πρόβλημα, Μπορούμε να διακρίνουμε τέσσερις διαφορετικούς τύπους εκχώρησης

τιμής μιας μεταβλητής, οι οποίοι ενέχουν διαφορετικού επιπέδου διδακτικά προβλήματα [Κόμης, 2001]:

### 1. Εκχώρηση σταθερής τιμής

π.χ.  $a:=15$ ,  $list:="word"$ ,  $test:=true$

### 2. Εκχώρηση τιμής μετά από υπολογισμό

π.χ.  $a:=15+4$ ,  $y:=4*x+5$

Οι δύο αυτές περιπτώσεις δεν εμφανίζουν ιδιαίτερα διδακτικά προβλήματα. Δε συμβαίνει όμως το ίδιο για τις επόμενες περιπτώσεις.

### 3. Αντικατάσταση (duplication)

π.χ.  $x:=y$

### 4. Συσσώρευση (accumulation)

π.χ.  $x:=x+1$ ,  $sum:=sum+n$ ,  $expr:=expr*n$

Η τέταρτη περίπτωση, όπου παρουσιάζεται στην πλήρη διάστασή της η μεταβλητή στην πληροφορική, είναι εκείνη που προκαλεί τις περισσότερες διδακτικές δυσκολίες γιατί έρχεται σε αντίθεση με τα νοητικά μοντέλα των μαθητών στα μαθηματικά. Στον προγραμματισμό μπορούμε να διακρίνουμε **δύο τύπους μεταβλητών** σε σχέση με τα νοητικά μοντέλα που χρησιμοποιούν οι μαθητές για να σχεδιάσουν τη λύση ενός προβλήματος. Έχουμε λοιπόν τις **ενδογενείς και τις εξωγενείς** μεταβλητές. Εξωγενείς είναι οι μεταβλητές που αναπαριστούν διασαφηνισμένα δεδομένα του προβλήματος ενώ ενδογενείς είναι οι μεταβλητές που χρησιμοποιούνται στον προγραμματισμό και ενώ δεν έχουν σημασιολογικό αντίστοιχο είναι απαραίτητες από στο πρόγραμμα. Έρευνες έχουν δείξει ότι το διδακτικό πρόβλημα είναι με τις ενδογενείς μεταβλητές αφού για να τις αναπαραστήσουν οι μαθητές πρέπει να έχουν γνώση της εσωτερικής λειτουργίας του υπολογιστή.

Ένα σημαντικό επίσης διδακτικό πρόβλημα που απαντάται κυρίως στους αρχάριους προγραμματιστές είναι ότι ξεχνούν πολύ συχνά τις απαραίτητες **αρχικοποιήσεις μεταβλητών**. Η αρχικοποίηση απαιτεί από τους μαθητές να κάνουν την υπόθεση ότι το σύστημα είναι ήδη σε μια αρχική κατάσταση. Ειδικά οι αρχικοποιήσεις στις δομές επανάληψης που είναι απαραίτητες είναι ιδιαίτερο πρόβλημα μιας και δεν χρειάζεται κάποια αρχικοποίηση σε προβλήματα που λύνονται με μη προγραμματιστικό τρόπο.

## 5.2. Δομές ελέγχου και η έννοια της επιλογής

Σε ένα πρόγραμμα που χρησιμοποιεί την δομή επιλογής χάνεται η σειριακή εκτέλεση των εντολών και το πρόγραμμα πρέπει να ακολουθήσει διαφορετικό δρόμο για άλλες εισόδους. Έτσι οι σειριακές αναπαραστάσεις του μαθητή θα έρθουν σε αντίθεση με τις σημασιολογικές και συντακτικές ιδιότητες του προγράμματος.

Οι μαθησιακές δυσκολίες που προκύπτουν συνδέονται με τα ακόλουθα σημεία [Κόμης, 2001]:

- Λογικό περιεχόμενο των συνθηκών (συνδυασμός περιπτώσεων, αναλυτικότητα και αποκλειστικότητα, λογικές πράξεις σύζευξης, διάζευξης, άρνησης κτλ.).

- Συμβολικές αναπαραστάσεις αυτών των περιπτώσεων.
- Σημασιολογικές και συντακτικές ιδιότητες της δομής ελέγχου μέσα στη χρησιμοποιούμενη γλώσσα προγραμματισμού.
- Αλληλεπιδράσεις με τις αναπαραστάσεις της ακολουθιακής μορφής της εκτέλεσης.

### 5.2.1. Διδακτικά προβλήματα σχετικά με την έννοια της δομής ελέγχου

Η ανάλυση των σχέσεων ανάμεσα στο εννοιολογικό πεδίο και το έργο του προγραμματισμού με χρήση δομής ελέγχου δίνει μια σειρά από διδακτικά προβλήματα τα οποία και πρέπει να λαμβάνονται υπόψη κατά τη διδασκαλία του προγραμματισμού. Ο χώρος προβλημάτων που απαιτούν μια δομή ελέγχου μπορεί να οριστεί με πολύ γενικό τρόπο ως τα προβλήματα στα οποία μια επεξεργασία εξαρτάται από την τιμή μιας παραμέτρου [Rogalski, 1988]. Η εξάρτηση μπορεί να έχει τη μορφή μιας απλής υπό συνθήκη εντολής. Όπως και στις μεταβλητές Έχουμε και δω ενδογενείς και εξωγενείς συνθήκες. Οι ενδογενείς είναι αποτέλεσμα κάποιου υπολογισμού του προγράμματος και η εξωγενής είναι αποτέλεσμα κάποιας αλληλεπίδρασης με το χρήστη. Οι Έρευνες έχουν τα ίδια αποτελέσματα όπως και στις μεταβλητές δηλαδή οι μαθητές συναντούν μεγαλύτερη δυσκολία στις ενδογενής συνθήκες.

Μια συνθήκη μπορεί να περιέχει: σχέσεις που αφορούν μεταβλητές όπως οι αριθμοί ή οι χαρακτήρες επιτρέπουν να οδηγηθούμε σε οικεία σχήματα, η χρήση λογικών (Boolean) μεταβλητών ή δεδομένα τύπου χαρακτήρων (string), για τα οποία οι μαθητές δε διαθέτουν τις ίδιες πρότερες γνώσεις. απαιτεί νέες αναπαραστάσεις. Έτσι, έχει παρατηρηθεί ότι κατά τη συγγραφή ελέγχων σπάνια γίνονται λάθη τα οποία αφορούν απλούς αριθμητικούς ελέγχους (όπως αν  $M > 0$  ή όχι), αλλά σχετικά περισσότερα λάθη γίνονται με πιο σύνθετες αριθμητικές δομές (που αναπαριστούν διάρκεια) ή μεταβλητές μη αριθμητικού τύπου.

Ιδιαίτερες διδακτικές δυσκολίες παρουσιάζουν οι πολλαπλές δομές ελέγχου και οι εμφωλευμένες δομές μέσα σε άλλες. Τέλος, φαίνεται απαραίτητο να ληφθούν υπόψη οι αλληλεπιδράσεις ανάμεσα στη λογική της επεξεργασίας υπό συνθήκη, της σημασιολογικής και συντακτικής αναπαράστασης μέσα σε μια γλώσσα προγραμματισμού και των αναπαραστάσεων τις οποίες διαθέτει το υποκείμενο πάνω στις ιδιότητες εκτέλεσης από το πληροφορικό μέσο [Κόμης, 2001].

### 5.3. Έννοια της επανάληψης

Η οικοδόμηση της έννοιας της επανάληψης (iteration) εμπερικλείει τον προσδιορισμό των στοιχειωδών δράσεων ή κανόνων που πρέπει να επαναληφθούν και τη συνθήκη που προσδιορίζει το τέλος ή τη συνέχιση της επανάληψης [Rogalski, 1990]. Κάτω από αυτό το πρίσμα, τρεις είναι οι βασικές παράμετροι που απαιτούνται για την οικοδόμηση της επαναληπτικής διαδικασίας: οικοδόμηση και σχεδιασμός της επεξεργασίας, ανάπτυξη δηλαδή της έκφρασης των εντολών που θα επαναληφθούν (σώμα του βρόχου), προσδιορισμός της συνθήκης ελέγχου για το σταμάτημα της επανάληψης (έλεγχος), η οποία αφορά εκφράσεις που συναπαρτίζουν



το σώμα του βρόχου (και πρέπει συνεπώς να προσδιοριστεί η θέση τους μέσα σε αυτό), και προσδιορισμός της αρχικής κατάστασης των μεταβλητών του βρόχου (αρχικοποίηση) [Κόμης, Πάτρα 2001].

Πριν όμως μελετήσουμε τα επιμέρους διδακτικά προβλήματα, πρέπει να δούμε πώς οικοδομείται η επαναληπτική προσέγγιση επίλυσης προβλημάτων. Μια επαναληπτική διαδικασία συνιστά το πέρασμα από μια γνωστή αρχική κατάσταση σε μια τελική ζητούμενη κατάσταση και οικοδομείται σε τέσσερις φάσεις [Arsac, 1980]:

- Καταρχήν γίνεται διατύπωση μιας επαγωγικής υπόθεσης, η οποία στην πραγματικότητα συνίσταται από την περιγραφή μιας ενδιάμεσης κατάστασης.
- Στη συνέχεια αναζητείται ο τρόπος περάσματος από μια κατάσταση στην επόμενη της, περιγράφεται δηλαδή όλη η ακολουθία των εντολών που επιτρέπουν αυτό το πέρασμα.
- Στο επόμενο βήμα διατυπώνεται η συνθήκη ελέγχου για το τέλος της επανάληψης, δηλαδή τότε η ενδιάμεση κατάσταση ταυτίζεται με την τελική κατάσταση.
- Τέλος, αναζητείται από πού αρχίζει η επαναληπτική διαδικασία, βρίσκεται δηλαδή μια περίπτωση όπου ισχύει προφανώς η επαγωγική υπόθεση. Αυτή δεν είναι άλλη από την περίπτωση όπου γνωρίζουμε όλες τις τιμές των μεταβλητών που εμπλέκονται στην επανάληψη, και συνήθως είναι η γνωστή αρχική κατάσταση.

Συμπερασματικά, η επαναληπτική διαδικασία προϋποθέτει την οικοδόμηση γνωστικών δεξιοτήτων υψηλού επιπέδου και νοητικών στοιχείων ελέγχου ενός μη καθορισμένου συνόλου πράξεων. Στο μαθηματικό επίπεδο, το βασικό εργαλείο ελέγχου είναι η **μαθηματική επαγωγή** (στο σημείο αυτό η επανάληψη συναντά την αναδρομή, βλέπε παρακάτω) με τη βοήθεια της οποίας εγκαθιδρύεται η κληρονομικότητα της ιδιότητας  $P(n)$  ή του συνόλου πράξεων  $P(n)$  στο πέρασμα από το  $n$  στο  $n+1$ , η οποία θα ισχύει όσες φορές είναι απαραίτητο.

### 5.3.1. Διδακτικά προβλήματα σχετικά με την έννοια της επανάληψης

Όταν ο αριθμός των επαναλήψεων δεν είναι γνωστός τότε υπάρχουν δυο τρόποι για την επίλυση του προβλήματος. Ο πρώτος είναι με τη χρήση της δομής ΟΣΟ συνθήκη ΕΤΤΑΝΕΛΑΒΕ και ο δεύτερος είναι με τη χρήση της δομής ΑΡΧΗ ΕΤΤΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ ΟΤΟΥ συνθήκη.

Καθένας από τους παραπάνω τρόπους δεν παρουσιάζει τα ίδια διδακτικά προβλήματα ούτε οικοδομείται με την ίδια ευκολία. Οι έρευνες που έχουν γίνει πάνω στα νοητικά μοντέλα των μαθητών δείχνουν ότι αναπαραστάσεις που διαθέτουν σχετικά με την επανάληψη έχουν κάποια συγκεκριμένη δομή: περιγραφή της δράσης, μετρητής επαναλήψεων, προσδιορισμός της επανάληψης και, τέλος, συνθήκη ελέγχου. Η συνθήκη ελέγχου αποτελεί διδακτικό πρόβλημα μιας και δεν εμφανίζεται στην επεξεργασία με το χέρι ή στις συνήθειες (μη προγραμματιστικές) δραστηριοτήτες των μαθητών, οπότε και υπονοείται. Σύμφωνα λοιπόν με τα προηγούμενα, ο τύπος επαναληπτικής δομής που ταιριάζει περισσότερο με τις αρχικές ιδέες των μαθητών είναι ο ΑΡΧΗ ΕΤΤΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ ΟΤΟΥ συνθήκη ενώ, αντίθετα, η δομή

ΟΣΟ συνθήκη ΕΠΑΝΕΛΑΒΕ ακολουθεί αντίθετη πορεία συλλογισμού, και γι' αυτό το λόγο εμφανίζει περισσότερα προβλήματα κατανόησης. Επιπλέον, κατά την υλοποίηση της δομής ΟΣΟ συνθήκη ΕΠΑΝΕΛΑΒΕ η δράση μπορεί να μην εκτελεστεί ποτέ, πράγμα που έρχεται σε αντίθεση με την αναπαράσταση της επαναληπτικής διαδικασίας που αρχικά διαθέτουν οι μαθητές. Σε αντίθεση με τη δομή ΑΡΧΗ ΕΠΑΝΑΛΗΨΗΣ....ΜΕΧΡΙΣ ΟΤΟΥ συνθήκη που εκτελείται τουλάχιστον μία φορά.

Ένα άλλο σημαντικό διδακτικό πρόβλημα τίθεται από την ίδια την εγκαθίδρυση του σώματος του βρόχου. Ο μαθητής οφείλει να προσδιορίσει, αφενός, τις εμπλεκόμενες μεταβλητές και, αφετέρου, τη σχέση ανάμεσά τους, η οποία πρέπει να διατηρηθεί σε όλη τη διάρκεια του βρόχου [Rogalski, 1990].

Όταν είναι εκ των προτέρων γνωστός ο αριθμός των επαναλήψεων, χρησιμοποιείται η δομή ΓΙΑ μεταβλητή ΑΠΟ τιμή ΜΕΧΡΙ τιμή ΜΕ ΒΗΜΑ τιμή ΕΠΑΝΕΛΑΒΕ. Σε καθαρά συντακτικό επίπεδο, είναι πιο απλή από τη ΑΡΧΗ ΕΠΑΝΑΛΗΨΗΣ ....ΜΕΧΡΙΣ ΟΤΟΥ, αφού δεν ελέγχει καμιά συνθήκη, ενώ παράλληλα είναι πιο εύκολο να δούμε τον αριθμό των επαναλήψεων που πραγματοποιεί. Τα Διδακτικά προβλήματα που προκύπτουν είναι παρόμοια με τις υπόλοιπες δομές επανάληψης εκτός του ότι εδώ δεν έχουμε συνθήκη για έλεγχο.

## 6. Πειραματικό εκπαιδευτικό περιβάλλον

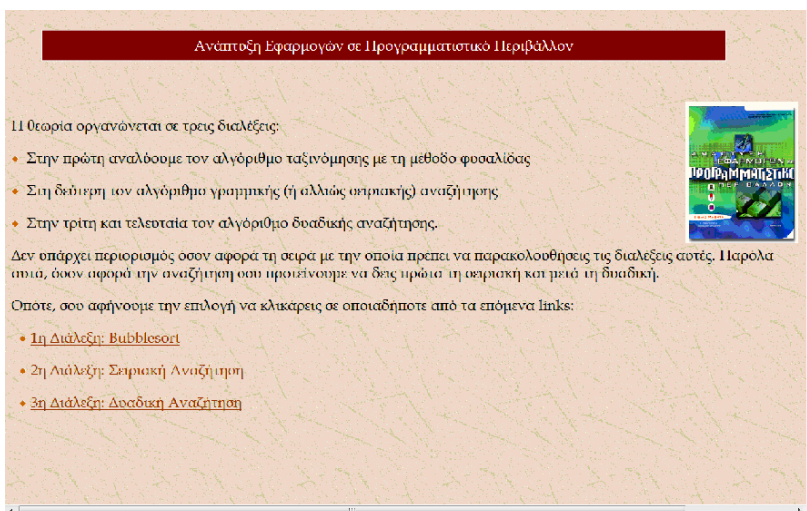
Το περιβάλλον που θα περιγράψουμε είναι μια ιστοσελίδα σχεδιασμένη για την διδασκαλία του μαθήματος: Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον (ΑΕΠΠ) που διδάσκεται στην τρίτη λυκείου.

Η αρχική σελίδα που εμφανίζεται στον χρήστη στο πάνω μέρος της εμφανίζει έξι links :

1. Θεωρητικό υπόβαθρο
2. Ασκήσεις
3. Java Applets
4. Online tests
5. Πηγές πληροφόρησης

Ο χρήστης προτρέπεται να ξεκινήσει από το απαραίτητο Θεωρητικό υπόβαθρο.

Πατώντας πάνω στο link Θεωρητικό υπόβαθρο εμφανίζεται η παρακάτω σελίδα που δίνει στο χρήστη την επιλογή να επιλέξει ποια από τις τρεις διδακτικές ενότητες θέλει να παρακολουθήσει.



Anáptuxh Epharmogón se Programmatistiko Perivállon

Η θεωρία οργανώνεται σε τρεις διαλέξεις:

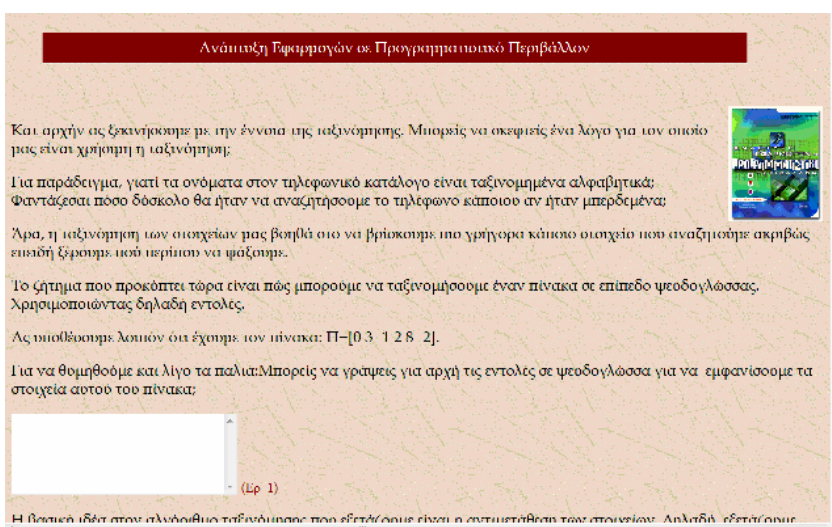
- Στην πρώτη αναλύουμε τον αλγόριθμο ταξινόμησης με τη μέθοδο φυσάλιδας
- Στη δεύτερη τον αλγόριθμο γραμμικής (ή αλλιώς σειριακής) αναζήτησης
- Στην τρίτη και τελευταία τον αλγόριθμο δοαδικής αναζήτησης.

Δεν υπάρχει περιορισμός όσον αφορά τη σειρά με την οποία πρέπει να παρακολουθήσεις τις διαλέξεις αυτές. Παρόλα αυτά, όσον αφορά την αναζήτηση σου προτείνουμε να δεις πρώτα τη σειριακή και μετά τη δοαδική.

Οποτε, σου αφήνουμε την επιλογή να κλικάρεις σε οποιαδήποτε από τα επόμενα links:

- [1η Διάλεξη: Bubblesort](#)
- [2η Διάλεξη: Σειριακή Αναζήτηση](#)
- [3η Διάλεξη: Δοαδικη Αναζήτηση](#)

Επιλέγοντας την πρώτη διάλεξη το λογισμικό μας παραπέμπει στη παρακάτω σελίδα:



Anáptuxh Epharmogón se Programmatistiko Perivállon

Και αρχίνας ξεκινάσουμε με την έννοια της ταξινόμησης. Μινορείς να σκεφτείς ένα λόγο για τον οποίο μας είναι χρήσιμη η ταξινόμηση;

Για παράδειγμα, γιατί τα ονόματα στον τηλεφωνικό κατάλογο είναι ταξινομημένα αλφαβητικά; Φαντάζεσαι ποσο δύσκολο θα ήταν να αναζητήσουμε το τηλέφωνο κάποιου αν ήταν μπερδεμένα;

Άρα, η ταξινόμηση των στοιχείων μας βοηθά στο να βρούμε πιο γρήγορα κάποιο στοιχείο που αναζητούμε ακριβώς επειδή έχουμε ποσ περιποσ να ψάξουμε.

Το ζήτημα που προκύπτει τώρα είναι πώς μπορούμε να ταξινομήσουμε έναν πίνακα σε επίπεδο ψευδογλώσσας, χρησιμοποιώντας δηλαδή εντολές.

As υποθέσουμε λοιπόν ότι έχουμε τον πίνακα: Π=[0 3 1 2 8 2].

Για να το μοινοδοίμε και λίγο τα παλιά: Μπορείς να γράψεις για αρχή τις εντολές σε ψευδογλώσσα για να εμφανίσουμε τα στοιχεία αυτού του πίνακα;

Α

(Σφ 1)

Η βασική ιδέα στον αλγόριθμο ταξινόμησης που εφετάσαμε είναι η αντιμετάθεση των στοιχείων. Αλλάζα εφετάσαμε

Σε αυτό το σημείο γίνεται η αφόρμηση και ένας μικρός έλεγχος προαπαιτούμενων γνώσεων. Παρακάτω γίνεται η παρουσίαση του αλγόριθμου δείχνοντας κάποια στιγμιότυπα από την εκτέλεσή του και γίνονται ερωτήσεις με κατάλληλα πλαίσια που ο χρήστης μπορεί να συμπληρώσει την απάντησή του.

**1ο Βήμα**

Αρχίζουμε από το τελευταίο στοιχείο του πίνακα, συγκρίνουμε τα στοιχεία ανά 2 και το 'ελαφρύτερο' πηγαίνει προς τα πάνω.

n	n	n	n	n	->
3	3	3	3	2	0
1	1	1	2	3	3
2	2	2	1	1	1
8	->	>	>	>	>
->	8	8	8	8	8

Παρατήρησε εδώ ότι σε κάθε περίπτωση έχουμε αντιμετάθεση. Μπορείς να φανταστείς γιατί συμβαίνει αυτό; Η απάντηση είναι προφανής. Εφόσον το -2 είναι το μικρότερο στοιχείο του πίνακα δικαιούται να ανεβεί στην πρώτη θέση (είναι η ελαφρύτερη φουσαλίδα) άρα σε κάθε σύγκριση θα ανεβάνει κι από μια θέση.

Και τώρα μια άλλη ερώτηση: Τι θα γινόταν αν σε κάποια σύγκριση οι 2 δεν ήταν οι μικρότερο στοιχεία; Θα υπήρχε ίσοιξη τις συγκρίσεις.

Συμπλήρωσε την απάντησή σου εδώ:

 (Fr\_2)

Η απάντηση είναι όχι. Αλλά ο πίνακας θα διατηρούσε την ίδια μορφή και θα συγκρίναμε το επόμενο ζεύγος. Αυτό που θέλω να σου πω είναι ότι η αντιμετάθεση δεν είναι απαραίτητο να συμβαίνει σε κάθε σύγκριση.

**2ο Βήμα**

Αφού παρουσιαστούν όλα τα βήματα εκτέλεσης του αλγόριθμου για τον συγκεκριμένο πίνακα το λογισμικό συνεχίζει με ερωτήσεις που αφορούν την κωδικοποίηση του αλγορίθμου της φουσαλίδας. Η φουσαλίδα μπορεί να θεωρηθεί ένα σύνολο από υποπροβλήματα τα οποία καλείται να λύσει ο χρήστης συμπληρώνοντας τα κατάλληλα πλαίσια όπως φαίνεται παρακάτω.

Πάμε τώρα να κωδικοποιήσουμε τον αλγόριθμο ταξινόμησης φουσαλίδας με εντολές της ψευδογλώσσας:

Υποθέτουμε ότι έχουμε έναν πίνακα Π, μεγέθους N, που αποτελείται πραγματικούς αριθμούς.

Χρειαζόμαστε μια δομή επανάληψης για τα βήματα του αλγορίθμου. Ποιά θα είναι αυτή;

Συμπλήρωσε την απάντησή σου εδώ:

 (Fr\_9)

Γνώσης, σε κάθε ένα από τα βήματα πρέπει να γίνεται μια σειρά από συγκρίσεις. Οι συγκρίσεις αυτές θα πρέπει να γίνονται μέσα σε μια επανάληψη. Άρα, θα έχουμε επωφελεμένη επανάληψη! Συμφωνείς; Ποια θα είναι η κωδικοποίηση του αλγορίθμου μέχρι αυτό το στάδιο;

Συμπλήρωσε την απάντησή σου εδώ:

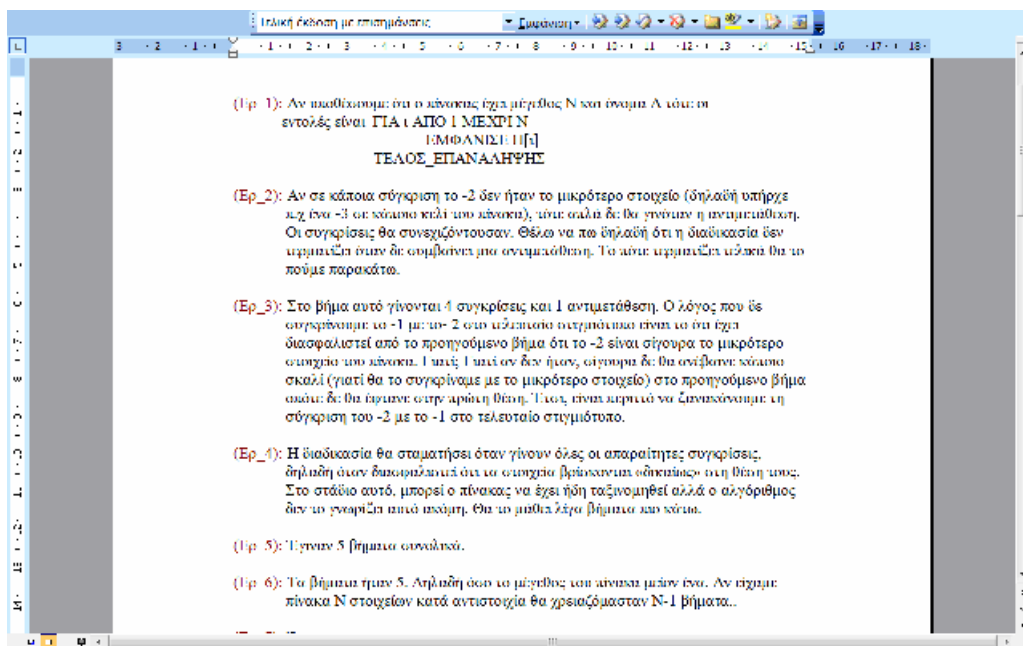
 (Fr\_10)

Έπειτα, πρέπει να γράψουμε τις εντολές για να συγκρίνουμε τα εκάστοτε 2 στοιχεία του πίνακα. Φαντάζομαι πως συμφωνείς ότι χρειαζόμαστε μια δομή επιλογής. Πως εμπλοστίζεται τώρα ο αλγόριθμος;

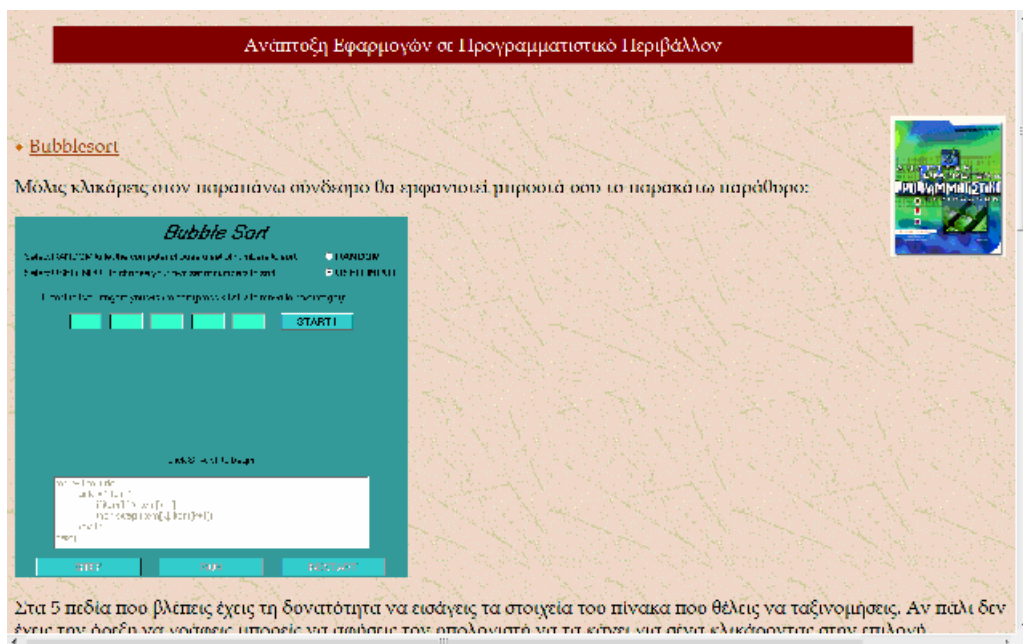
Συμπλήρωσε την απάντησή σου εδώ:

Κατ' αυτόν τον τρόπο ο χρήστης οικοδομεί την κωδικοποίηση της φουσαλίδας μέσα από την αλληλεπίδραση του με το λογισμικό. Πριν το τέλος της σελίδας υπάρχει ένας σύνδεσμος που βρίσκονται οι απαντήσεις έτσι ώστε να μπορεί ο μαθητής να ελέγξει

την ορθότητα των απαντήσεών του. Πατώντας στις απαντήσεις εμφανίζεται ένα αρχείο .doc όπως αυτό που φαίνεται στην παρακάτω εικόνα:



Στο κάτω μέρος της σελίδας ο μαθητής παρακινείται να εξετάσει ένα java applet για τον αλγόριθμο της φυσαλίδας. Κάνοντας αυτή την επιλογή το λογισμικό μας παραπέμπει στην πιο κάτω σελίδα:



Εδώ δίνονται κάποια στιγμιότυπα από την εφαρμογή και λεπτομέρειες για τον τρόπο λειτουργίας. Λίγο παρακάτω στην ίδια σελίδα απευθύνονται στο μαθητή κάποια ερωτήματα που καλείται να απαντήσει χρησιμοποιώντας το applet, ενώ δίνονται και οι απαντήσεις σε ξεχωριστό αρχείο .doc.

```

#include <string.h>
int main() {
    int N;
    int arr[10];
    int i;
    for (i = 0; i < N; i++)
        arr[i] = i + 1;
    bubbleSort(arr, N);
    return 0;
}

```

Παιτώντας διαδοχικά το πλήκτρο STTP εμφανίζονται όλα τα στοιχεία του αλγορίθμου μέχρι αυτός να τερματίσει.

Για αρχή, γιατί ξεκινάει με τη σύγκριση των πρώτων 2 στοιχείων του πίνακα αντί των 2 τελευταίων που κάνουμε εμείς; Εννοώ ποιες εντολές στον ψευδοκώδικα το καθορίζουν αυτό;(Lp 1)

Μπορείς να σκεφτείς όμως να δώσεις έναν πίνακα στο applet και να μη γίνει καμιά αντιμετάθεση;(Ep\_2)

Μπορείς να σκεφτείς ώστε να δώσεις έναν πίνακα στο applet και να γίνεται πάντα αντιμετάθεση (σε κάθε σύγκριση εννοώ);(Lp 3)

Άσε τον υπολογιστή να συμπληρώσει τον πίνακα για σένα και μετά πάτα START. Κάθε φορά, προτού πατήσεις STTP προσπάθησε να προβλέψεις τι θα κάνει ο αλγόριθμος και μετά άσε το applet να εμβεβιώνει (ελιγίζω) την εμφάνισή σου.

Αφού τελειώσεις με ένα παράδειγμα, μπορείς να ξεκινήσεις ένα καινούριο πατώντας το RESTART.

Μπορείς να δεις τις απαντήσεις στα παραπάνω ερωτήματα αλλά όχι χωρίς να απαντήσεις πρώτα από μόνος σου! Μπες στον κόπο να σκεφτείς τα παραπάνω προτού καταβύσεις το αρχείο των απαντήσεων. Ηιστοσελίδα με θα κερδίσεις αρκετά αν προβληματιστείς λίγο παραπάνω. ♦ [Απαντήσεις](#)

Πατώντας πάνω στο σύνδεσμο Απαντήσεις εμφανίζει το παρακάτω αρχείο:

(Ep\_1): Αντί η εκδοχή του bubblesort επιλέγει τα πρώτα στοιχεία για σύγκριση (αντί για τα τελευταία και καλύτερα: εμείς). Η εντολή `arr[j]-1` το τι θα καθορίσει να μην θα ξεκινήσει από το 1ο στοιχείο του πίνακα (αντί για το N-ιστό που επιλέγουμε εμείς). Η λίστα είναι συμμετρική ως προς τη θέση μας. Δεν κάνει βίβλιον να είναι ορθή.

(Ep\_2): Για να μη γίνει καμιά αντιμετάθεση θα πρέπει ο πίνακας να είναι ήδη ταξινομημένος. Άρα, αρκεί να δώσω έναν οποιοδήποτε ήδη ταξινομημένο πίνακα πχ [1 2 3 4 5] ή [40 50 80 100 300]. Δοκιμάστε κι εσύ έναν!

(Ep\_3): Για να γίνεται σε κάθε σύγκριση αντιμετάθεση, σημαίνει ότι ο πίνακας είναι τελείως αταξινόμητος! Δηλαδή, δεν υπάρχει ούτε ένα ζεύγος στοιχείων που να είναι ταξινομημένο κατά αύξουσα σειρά. Άρα, είναι έναν πλήρως αταξινόμητο πίνακα πχ [5 4 3 2 1]. Δοκιμάστε κι εσύ έναν!

Ο εκπαιδευόμενος αφού βεβαιωθεί ότι έχει κατανοήσει τον αλγόριθμο καλείται να λύσει ένα quiz για να εξετάσει τις γνώσεις του. Επιλέγοντας το σύνδεσμο online tests εμφανίζεται η παρακάτω σελίδα με τρία τεστ:

Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον

- ♦ [Quiz 1](#)
- ♦ [Quiz 2](#)
- ♦ [Quiz 3](#)



Αφού τελειώσεις με τα quiz, είναι καλός να ασχοληθείς με διάφορα προβλήματα στα οποία βρίσκουν εφαρμογή οι αλγόριθμοι που αναλύσαμε ως τώρα.

Αν σε ενδιαφέρει κάνε κλικ εδώ > ♦ [Ασκήσεις](#)

Τα quiz έχουν την παρακάτω μορφή:

First name:	<input type="text"/>	Last name:	<input type="text"/>
1	Μπορούμε να εφαρμόσουμε τον αλγόριθμο φυσικής σε έναν αλφαριθμητικό πίνακα		
	<input type="radio"/> False <input type="radio"/> True		
2	Όταν ψάχνουμε σε ένα τηλεφωνικό κатάλογο χρησιμοποιούμε τη σειριακή μέθοδο αναζήτησης.		
	<input type="radio"/> True <input type="radio"/> False		
3	Η ταξινόμηση των στοιχείων ενός πίνακα πρέπει να γίνεται πάντα πριν από την αναζήτηση.		
	<input type="radio"/> False <input type="radio"/> True		
4	Για να εφαρμοστεί η μέθοδος της σειριακής αναζήτησης είναι απαραίτητο τα στοιχεία να είναι ταξινομημένα		
	<input type="radio"/> True <input type="radio"/> False		

Αν ο μαθητής επιλέξει το σύνδεσμο Ασκήσεις εμφανίζεται η παρακάτω σελίδα:

**Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον**

Ήρθε η ώρα να γράψεις το πρώτο σου πραγματικό πρόγραμμα! Και λέω πραγματικό γιατί ως τώρα όλα τα έλυνες στο χαρτί. Τώρα θα έχεις την ευκαιρία να δεις το πρόγραμμά σου να "τρέχει" όπως λένε συνήθως οι προγραμματιστές.

Για αρχή θα λύσουμε ένα πρόβλημα παρέα, περισσότερο για να σου δείξω πως δουλεύει το πρόγραμμα που θα χρησιμοποιείς για να γράφεις τα δικά σου προγράμματα.

Κατ'αρχήν κατεβάζεις το παρακάτω setup και το εγκαθιστάς στον υπολογιστή σου.

• [Setup Γλωσσολέχεια](#)

Για να το ανοίξεις πηγαίνεις κλασικά Έναρξη→Προγράμματα → Γλωσσολέχεια (έτσι ονομάζεται το πρόγραμμα).

Ας υποθέσουμε λοιπόν, ότι έχουμε το παρακάτω πρόβλημα:

Μια βιβλιοθήκη διαθέτει 10 βιβλία. Να γράψετε πρόγραμμα το οποίο:

- θα διαβάζει το έτος έκδοσης όλων των βιβλίων της βιβλιοθήκης σε έναν πίνακα Λ.
- θα ζητάει από το χρήστη να εισάγει ένα έτος
- θα αναζητεί αν υπάρχει έτος και ένα βιβλίο που εκδόθηκε το έτος που έδωσε ο χρήστης.

Αφού γράψουμε το πρόγραμμά, πηγαίνουμε Πρόγραμμα→Έλεγχος Σφαλμάτων και εμφανίζονται τα συντακτικά λάθη

Εδώ δίνεται ένας σύνδεσμος για το εκπαιδευτικό λογισμικό Γλωσσολέχεια το οποίο μπορεί να εγκαταστήσει στον υπολογιστή του. Παρακάτω στην ίδια σελίδα δίνονται στιγμιότυπα επίλυσης ενός προβλήματος με τη Γλωσσολέχεια όπως δείχνει η παρακάτω εικόνα:

Αφού γράψουμε το πρόγραμμα, πηγαίνουμε Πρόγραμμα → Έλεγχος Σφαλμάτων και εμφανίζονται τα συντακτικά λάθη του κώδικα. Θα πρέπει να θυμάσαι από το 6ο κεφάλαιο ότι ο μεταγωγιστής δεν αφήνει ένα πρόγραμμα να εκτελεστεί παρά μόνο όταν είναι συντακτικά ορθό.

```

1 ΠΡΟΓΡΑΜΜΑ ΒΙΒΛΙΟΘΗΚΗ
2 ΠΡΟΓΡΑΜΜΑΤΙΣΤΗΣ
3 ΑΡΧΗ:ΤΡΕΞΕ: ΕΙΣ[1],1,1000
4 ΔΙΑΒΑΣΕ Βιβλίων
5 ΑΣΟΣ
6 ΤΡΕΞΕ: 2011,1 ΜΑΡΤΟΣ 10
7 ΣΤΡΩΣΕ Τέτος να είναι τα έτη των βιβλίων του βιβλίου
8 ΔΙΑΒΑΣΕ Ε[1]
9 ΤΡΩΣΕ ΒΙΒΛΙΟΘΗΚΗ
10 ΣΤΡΩΣΕ Τέτος ένα έτος
11 ΔΙΑΒΑΣΕ Έτος
12 ΒΟΗΘΗΣΕ Έτος
13 Ε
14 ΚΑΙ ΕΠΙΣΤΡΕΦΕΙΝΤΕΣ ΕΚΕΙΝΟΙ ΔΙΑΒΑΣΤΕ
15 ΑΝ ΔΙΑΒΑΣΕ ΤΕΤΟΣ
16 ΠΡΟΒΛΕΨΕ Ακρίβεια
17 ΣΤΡΩΣΕ Τέτος βιβλίο που εκδόθηκε αυτό το έτος
18 ΚΡΑΤΩΣΕ Ε
19 Ε
20 ΤΡΩΣΕ ΒΙΒΛΙΟΘΗΚΗ
21 ΑΝ ΕΠΙΣΤΡΕΦΕΙΝΤΕΣ ΤΟΤΕ
22 ΣΤΡΩΣΕ Έτος βιβλίο βιβλίο που εκδόθηκε αυτό το έτος
23 ΤΡΩΣΕ Ε
24 ΤΡΩΣΕ ΒΙΒΛΙΟΘΗΚΗ
    
```

Αφού εξαλείψουμε τα συντακτικά λάθη πηγαίνουμε Πρόγραμμα→Εκτέλεση και βλεπουμε αν το πρόγραμμά μας κάνει αυτά για τα οποία το φτιάξαμε(και χαίρομαστε) ή κάνει ότι να ναι οπότε υπάρχει πρόβλημα.

Για το παραπάνω πρόβλημα, η εκτέλεση έχει ως εξής:

```

Γλωσσολέχεια Εκτέλεση
0:00:00
0:00:00 κα κ.α.σ. εκτέλεση και 1 βιβλίου
1:00:00
0:00:00 κα κ.α.σ. εκτέλεση και 2 βιβλίου
1:00:00
0:00:00 κα κ.α.σ. εκτέλεση και 3 βιβλίου
1:00:00
0:00:00 κα κ.α.σ. εκτέλεση και 4 βιβλίου
1:00:00
0:00:00 κα κ.α.σ. εκτέλεση και 5 βιβλίου
1:00:00
    
```

Παράλληλα με τα στιγμιότυπα δίνονται και κάποιες εξηγήσεις για το πώς δουλεύει η Γλωσσομάθεια ενώ στο τέλος της σελίδας υπάρχει και ένας σύνδεσμος με τη λύση του παραδείγματος:

```

ΠΡΟΓΡΑΜΜΑ β.βελιόθλη
ΜΕΤΑΒΛΗΤΕΣ
ΑΚΕΡΑΙΕΣ: Α[10],ι,έτος
ΛΟΓΙΚΕΣ: βρέθηκε
ΑΞΙΟ
ΓΙΑ ι ΑΠΟ 1 ΜΕΧΡΙ 10
  ΓΡΑΨΕ 'άσσε το έτος έκδοσης του ',ι,' βιβλίου'
  ΔΙΑΒΑΣΕ Α[ι]
  ΣΕΛΟΣ ΕΠΑΝΑΛΗΨΗΣ
  ΓΡΑΨΕ 'άσσε ένα έτος'
  ΔΙΑΒΑΣΕ έτος
  βρέθηκε <-- ΨΕΥΔΗΣ
  ι <-- ι-1
  ΟΧΙ (βρέθηκε=ΨΕΥΔΗΣ) ΚΑΙ (ι<=10) ΕΠΑΝΑΛΑΒΕ
  ΑΝ Α[ι]=έτος ΤΟΤΕ
    βρέθηκε <-- ΑΛΗΘΗΣ
    ΓΡΑΨΕ 'Υπάρχει βιβλίο που εκδόθηκε αυτό το έτος'
  ΣΕΛΟΣ ΑΝ
  ι <-- ι+1
  ΣΕΛΟΣ ΕΠΑΝΑΛΗΨΗΣ
ΑΝ βρέθηκε=ΨΕΥΔΗΣ ΤΟΤΕ
  ΓΡΑΨΕ 'δεν υπάρχει βιβλίο που εκδόθηκε αυτό το έτος'
ΣΕΛΟΣ ΑΝ
ΣΕΛΟΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

```



Επιλέγοντας τη δεύτερη διάλεξη τη σειριακή αναζήτηση δηλαδή το λογισμικό εμφανίζει την παρακάτω σελίδα:

Κατ'αρχήν ας ξεκινήσουμε με την έννοια της αναζήτησης. Μια κλασική περίπτωση αναζήτησης είναι η αναζήτηση στον τηλεφωνικό κατάλογο. Για να φέρουμε και την κουβεντα μας λίγο πιο κοντά στο μάθημα, μπορείς να φανταστείς ότι ο τηλεφωνικός κατάλογος είναι ένας αλφαριθμητικός πίνακας ας πούμε 10.000 θέσεων όπου είναι αποθηκευμένα τα ονόματα όλων των κατοίκων ενός νομού. Φοιά ώρα θέλεις να βρεις στον πίνακα αυτό αν υπάρχει κάποιος το όνομα Παπαδόπουλος.

Η σειριακή αναζήτηση μας λέει έναν τρόπο να ψάξουμε στον πίνακα για τον Παπαδόπουλο. Συγκεκριμένα μας λέει:

1. Ξεκίνα με το 1ο στοιχείο του πίνακα
2. Είναι το στοιχείο αυτό που ψάχνεις;

Αν ναι, τότε σταμάτα εφόσον βρήκες τον Παπαδόπουλο.

Αν όχι, τότε εξέλαισε το επόμενο στοιχείο (δηλαδή το δεύτερο στη σειρά) και έλεγξε αν υπάρχει εκεί ο Παπαδόπουλος. Αν ναι σταμάτα, αλλιώς στο επόμενο κοκ.

Υπάρχουν 2 περιπτώσεις στις οποίες μπορεί να τερματίσει ο αλγόριθμος. Μήπως κατάλαβες ποιές είναι αυτές;

Συμπλήρωσε την απάντησή σου εδώ:

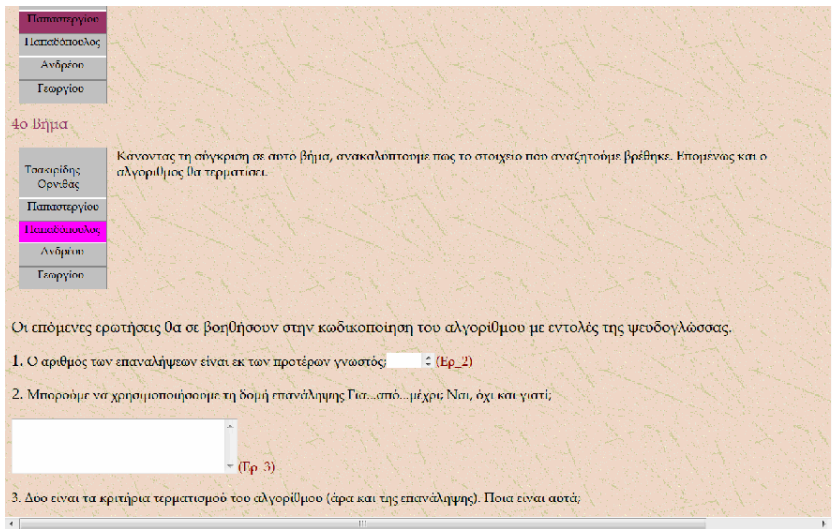
(Ερ\_1)

Θα δούμε τώρα με παραστατικό τρόπο όσα είπαμε πιο πάνω. Έστω ότι έχουμε τον παρακάτω πίνακα:

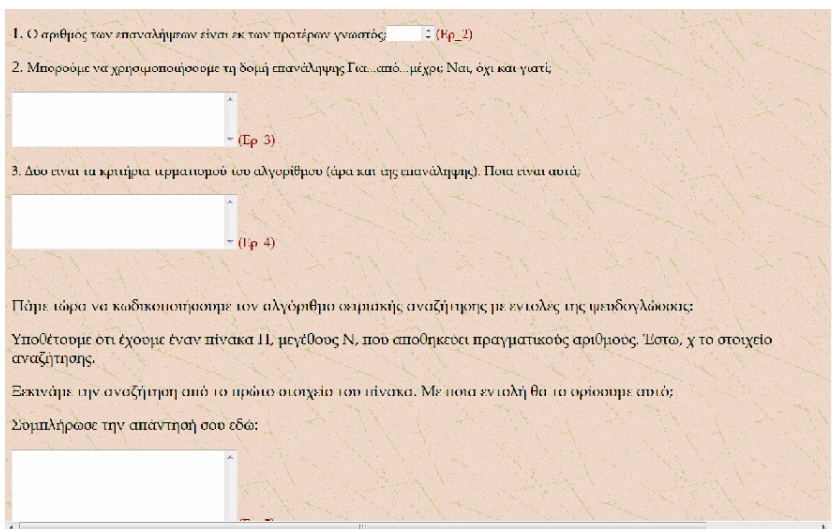
Τασιριόης

Εδώ γίνεται η εισαγωγή στο θέμα και δίνεται ένα παράδειγμα για την πρόκληση του ενδιαφέροντος του χρήστη. Παρακάτω παρουσιάζεται ο αλγόριθμος σε στιγμιότυπα τρεξίματος ενώ παράλληλα γίνονται και ερωτήσεις προς το χρήστη για την καλύτερη κατανόηση της λειτουργίας του αλγορίθμου. Ο χρήστης έχει την δυνατότητα να δίνει τις απαντήσεις του σε προκαθορισμένα πλαίσια όπως φαίνεται παρακάτω.

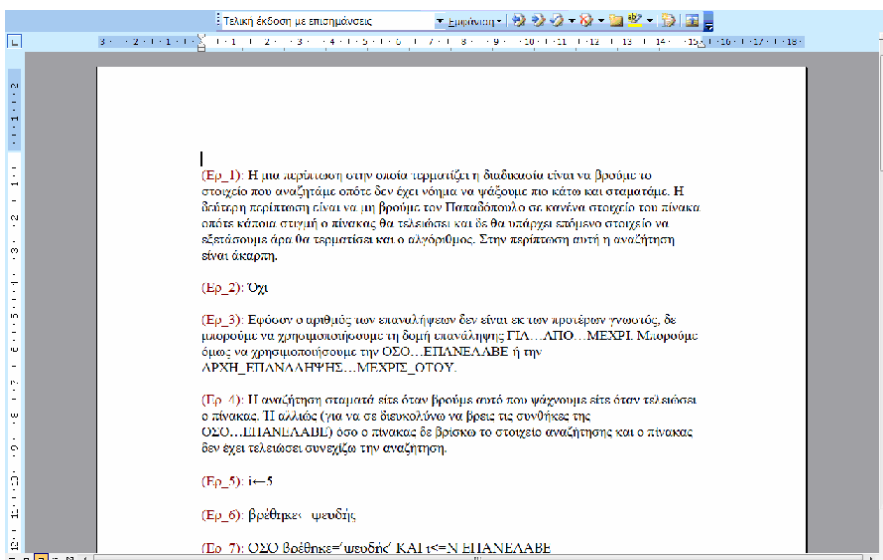




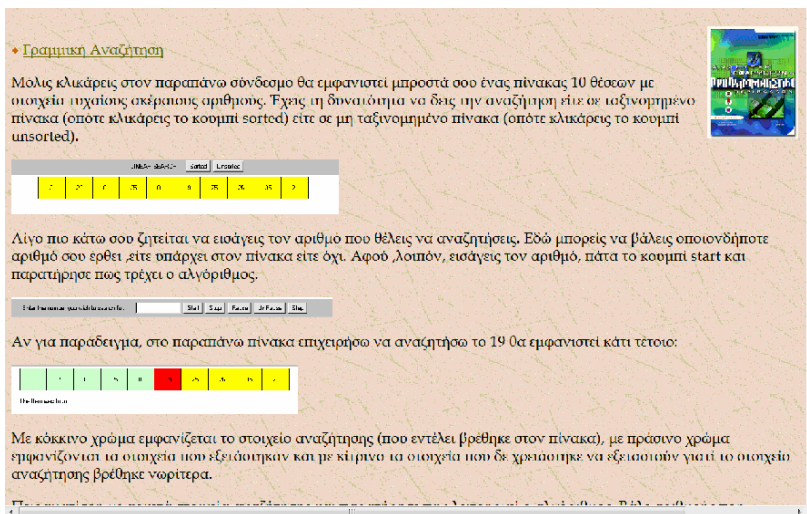
Παρακάτω στην ίδια σελίδα Αφού έχει γίνει η παρουσίαση του αλγορίθμου υπάρχουν ερωτήσεις και πλαίσια απαντήσεων που αφορούν την κωδικοποίηση του αλγορίθμου. Μέσα από αυτήν την διαδικασία των ερωτήσεων και απαντήσεων δίνεται στο χρήστη να «δημιουργήσει» μόνος του την κωδικοποίηση.



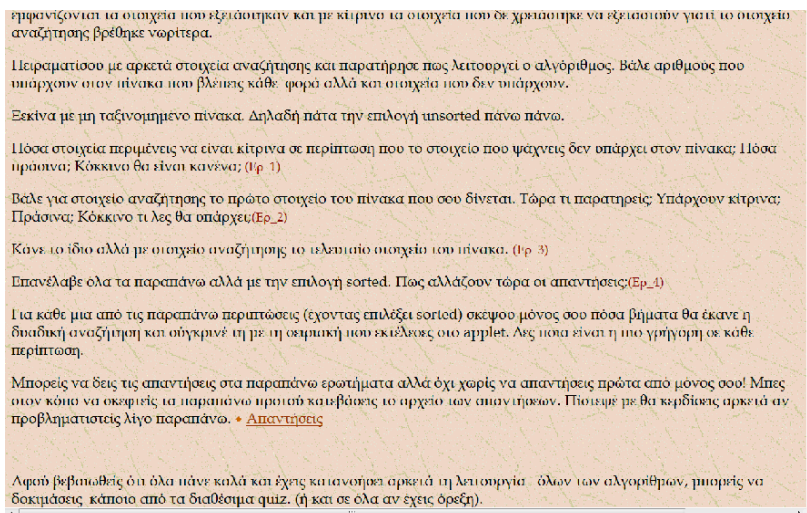
Ο χρήστης έχει τη δυνατότητα να δει τις σωστές απαντήσεις πατώντας στον σύνδεσμο Απαντήσεις όπως φαίνεται παρακάτω.



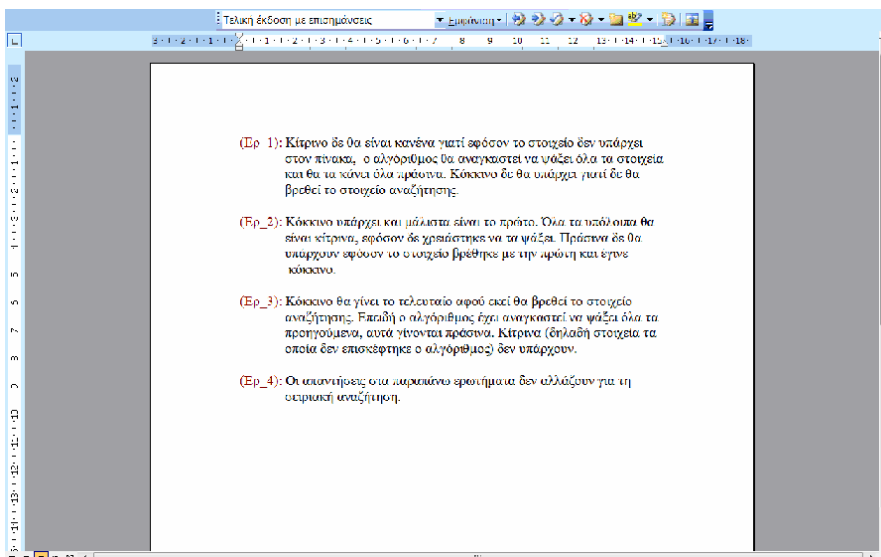
Μετά το τέλος της κωδικοποίησης υπάρχει ένας σύνδεσμος που οδηγεί σε ένα applet για την σειριακή αναζήτηση:



Εδώ δίνονται εξηγήσεις για την λειτουργία της εφαρμογής ενώ όπως φαίνεται παρακάτω υπάρχουν ερωτήματα στα οποία απαντάει ο μαθητής με χρήση του applet.



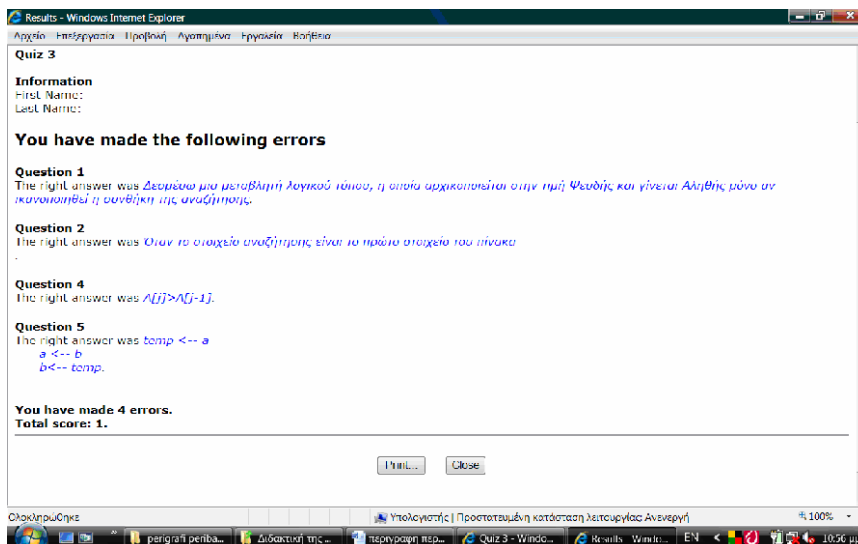
Για την επαλήθευση των απαντήσεων ο μαθητής μπορεί να κλικάρει πάνω στον σύνδεσμο Απαντήσεις και θα εμφανιστεί το παρακάτω αρχείο .doc με τις απαντήσεις.



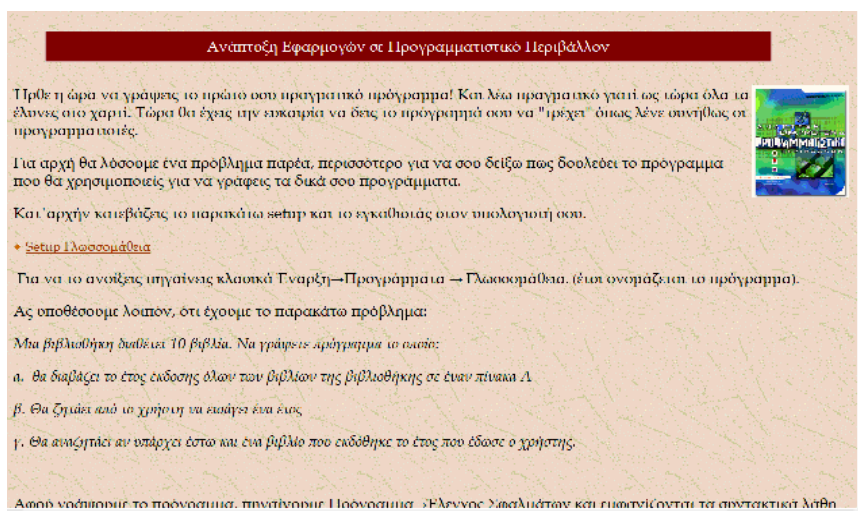
Έπειτα ο μαθητής παρακινείται να λύσει ερωτήσεις πολλαπλής επιλογής.



Αφού το συμπληρώσει έχει τη δυνατότητα να δει τη βαθμολογία του και στις λανθασμένες απαντήσεις να διαβάσει γιατί έκανε λάθος όπως φαίνεται παρακάτω.



Όπως και στην προηγούμενη διάλεξη μετά από τα quiz ο μαθητής παρακινείται να λύσει ασκήσεις με τη γλωσσομάθεια.



Αν ο μαθητής επιλέξει την τρίτη διάλεξη δηλαδή την δυαδική αναζήτηση θα εμφανιστεί η παρακάτω οθόνη:

Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον

Ο αλγόριθμος της δυαδικής αναζήτησης προτείνει έναν διαφορετικό και πιο σύντομο τις περισσότερες φορές τρόπο για να ψάχνουμε ένα στοιχείο σε έναν πίνακα.

Γίνει ακριβώς ο τρόπος με τον οποίο ψάχνουμε στον τηλεφωνικό κατάλογο. Παια, λοιπόν, ότι ψάχνουμε τον Παναδόσιου. Τι κάνουμε;

Ανοίγουμε τον κατάλογο περίπου στη μέση και παρατηρούμε σε ποιο γράμμα πέσαμε, έστω Μ. Εφόσον το Π είναι μετά το γράμμα Μ, απορρίπτουμε μεμιας όλα τα γραμμάτα πριν το Μ και ψάχνουμε μόνο στα εναπομείναντα. Έτσι, μειώνουμε πολύ τον αριθμό των στοιχείων που εξετάζουμε.

Θα δούμε τώρα με παρασιατικό τρόπο οσα είπαμε πιο πάνω. Παια ότι έχουμε τον παρακάτω πίνακα:

Τακτοίδη
Οφειόδη
Πασισιργίσι
Πασιδόσιουσι
Ανόμεσι
Γεωργίσι

Και αρχίν, μιμορούσι να εφαιρούσι δυαδική αναζήτιση σιον πίνακα σιού; Ναι, όχι και για τί;

Η διαδικασία εδώ είναι παρόμοια με τις άλλες δυο διαλέξεις με τη διαφορά ότι εδώ δεν γίνεται κωδικοποίηση του αλγόριθμου μιας και δεν διδάσκεται στην τρίτη λυκείου. Στο τέλος της σελίδας υπάρχει ένας σύνδεσμος για applet δυαδικής αναζήτησης για την οπτικοποίηση και κατανόηση του αλγόριθμου.

## 7. Σχέδια μαθήματος

### 7.1. Ταξινόμηση φυσαλίδας (bubblesort)

**Μάθημα:** Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον

**Τάξη:** Γ' Λυκείου

**Γνωστικό Πεδίο:** Ταξινόμηση φυσαλίδας (bubblesort)

#### Μεθοδολογία διδασκαλίας

Η μεθοδολογία που ακολουθείται στη διδασκαλία της ενότητας βασίζεται στην οπτικοποίηση του αλγόριθμου, την αλληλεπίδραση του μαθητή με αυτόν μέσω του λογισμικού και στις αρχές του εποικοδομητισμού. Ο μαθητής αποτελεί ενεργό μέλος στη διαδικασία της μάθησης αναλαμβάνοντας ο ίδιος πρωτοβουλίες. Σαν πρώτη φάση προκαλείται το ενδιαφέρον των μαθητών μέσα από παραδείγματα της καθημερινότητας, έπειτα ελέγχονται οι πρότερες γνώσεις των εκπαιδευόμενων και γίνεται αναδόμηση των ιδεών τους. Στη συνέχεια μέσα από την αλληλεπίδραση των μαθητών με το λογισμικό «χτίζεται» η καινούρια γνώση. Στο τέλος γίνεται εμπέδωση και εφαρμογή της νέας γνώσης.

#### Διδακτικοί Στόχοι

- 1) Οι μαθητές, στα πλαίσια αυτού του μαθήματος, θα πρέπει να καταλάβουν τη χρησιμότητα της ταξινόμησης ενός πίνακα και τότε την εφαρμόζουμε.
- 2) Να κατανοήσουν ποιοτικά τον τρόπο λειτουργίας του αλγορίθμου της φυσαλίδας
- 3) Να είναι σε θέση να εφαρμόζουν τον αλγόριθμο φυσαλίδας σε ασκήσεις όπου χρειάζεται ταξινόμηση

#### Φάσεις διδασκαλίας

##### **A) Έλεγχος προϋπάρχουν γνώσεων**

Οι απαραίτητες γνώσεις και δεξιότητες που πρέπει να έχουν οι μαθητές προκειμένου να διδαχτούν με επιτυχία τον αλγόριθμο της ταξινόμησης ξεκινώντας από τα πιο βασικά είναι:

- 1) Οι μαθητές να έχουν καταλάβει το νόημα της ανάθεσης τιμής σε μια μεταβλητή και τον τρόπο αντιμετάθεσης των τιμών δυο μεταβλητών με χρήση μια ενδιάμεσης προσωρινής μεταβλητής.

**Ερώτηση:**  $\alpha \leftarrow 10, \beta \leftarrow 20$ . Πως γίνεται η αντιμετάθεση των τιμών των  $\alpha, \beta$ ;

- 2) Να έχουν καλή γνώση του τρόπου λειτουργίας της δομής επιλογής `ΑΝ...ΑΛΛΙΩΣ...ΤΕΛΟΣ_ΑΝ` και ευχέρεια στη σύνταξη και τον χειρισμό της
- 3) Να γνωρίζουν την σύνταξη και τον τρόπο λειτουργίας της δομής επανάληψης «`ΓΙΑ...ΑΠΟ ...ΜΕΧΡΙ...ΜΕ_ΒΗΜΑ...`» με θετικό και αρνητικό βήμα.

**Ερώτηση:** Πόσες φορές θα εμφανιστεί η λέξη `HI` στο παρακάτω τμήμα αλγορίθμου;

```
α)ΓΙΑ i ΑΠΟ 0 ΜΕΧΡΙ 999 ΜΕ_ΒΗΜΑ 2
    ΕΜΦΑΝΙΣΕ 'HI'
```

ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ  
β)ΓΙΑ i ΑΠΟ 10 ΜΕΧΡΙ 1 ΜΕ\_ΒΗΜΑ -1  
ΕΜΦΑΝΙΣΕ 'ΗΙ'  
ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

4) Να ξέρουν τον τρόπο λειτουργίας των εμφουμένων δομών επανάληψης και τον αριθμό των επαναλήψεων που εκτελούνται σε αυτές.

**Ερώτηση:** Πόσες φορές θα εμφανιστεί η λέξη ΗΙ στο παρακάτω τμήμα αλγορίθμου;

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10  
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10  
ΕΜΦΑΝΙΣΕ 'ΗΙ'  
ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ  
ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ

5) Να είναι πλήρως εξοικειωμένοι με την έννοια του πίνακα και τον τρόπο προσπέλασης και ανάθεσης τιμών στα κελιά του πίνακα.

**Ερώτηση:**  $A = [4 \ 7 \ 2 \ 1 \ 9 \ 5]$ . Να γίνει αρχικοποίηση και εμφάνιση των στοιχείων του πίνακα A.

## **B) Αφόρμηση**

Η Αφόρμηση και η πρόκληση ενδιαφέροντος των μαθητών γίνεται μέσα στο λογισμικό και με κατάλληλες ερωτήσεις. Προκειμένου οι μαθητές να καταλάβουν τη χρησιμότητα της ταξινόμησης δίνεται ένα απλό παράδειγμα. Το παράδειγμα αναφέρεται στον τηλεφωνικό κατάλογο και την δυσκολία που θα είχαμε να αναζητήσουμε μια εγγραφή μέσα από ένα μη ταξινομημένο σύνολο.

## **Γ) Ανακάλυψη της νέας γνώσης μέσα από το λογισμικό**

Προτού δώσουμε τον αλγόριθμο ταξινόμησης φυσαλίδας σε μορφή εντολών, δείχνουμε βήματά του σε έναν αταξινομητο πίνακα. Υποθέτουμε έναν πίνακα ο οποίος δεν είναι ταξινομημένος. Στο λογισμικό παρουσιάζονται αναλυτικά τα βήματα του αλγορίθμου, και όλες οι αντιμεταθέσεις των στοιχείων μέσα από στιγμιότυπα τρεξίματός του. Παράλληλα με τα στιγμιότυπα τρεξίματος γίνονται κατάλληλες ερωτήσεις προς τον χρήστη και του δίνεται η δυνατότητα να καταγράψει τις απαντήσεις του. Στο τέλος υπάρχουν και οι σωστές απαντήσεις έτσι ώστε να μπορεί να ελέγξει την ορθότητα των δικών του απαντήσεων και ενδεχομένως ερχόμενος σε γνωστική σύγκρουση να αναθεωρήσει κάποιες λανθασμένες γνώσεις.

## **Δ) Εφαρμογή της νέας γνώσης**

Εφόσον έχουμε εξηγήσει «ποιοτικά» τα βήματα του αλγορίθμου μπορούμε να συνεργαστούμε με τους μαθητές ώστε να βρούμε τις εντολές του αλγορίθμου σε ψευδογλώσσα. Σπάμε τον αλγόριθμο σε μικρότερα προβλήματα όπως για παράδειγμα την αντιμετάθεση των στοιχείων ενός πίνακα, την σύγκριση τιμών δυο μεταβλητών κτλ. Έπειτα δίνεται η δυνατότητα στο μαθητή να λύνει τα επί μέρους προβλήματα καταγράφοντας την απάντησή του σε κατάλληλα πλαίσια που παρέχει το λογισμικό.

Έτσι σταδιακά «χτίζεται» η κωδικοποίηση του αλγορίθμου. Μέσα από αυτή τη διαδικασία δίνεται στον μαθητή η ευκαιρία να οικοδομήσει μόνος του τις γνώσεις και προωθούνται οι έννοιες του οικοδομιτισμού, και της αλληλεπίδρασης του υποκειμένου με το εκπαιδευτικό λογισμικό.

### **Ε) Εμπέδωση της νέας γνώσης**

Αφού έχει γίνει η ποιοτική παρουσίαση και κωδικοποίηση του αλγορίθμου, οι μαθητές έχουν εξοικειωθεί με την έννοια της ταξινόμησης. Προχωράμε στην εφαρμογή του αλγορίθμου σε ένα java applet που εξομοιώνει γραφικά τη λειτουργία της ταξινόμησης φυσαλίδας. Το applet δίνει τη δυνατότητα στον εκπαιδευόμενο να αλλάξει τις τιμές του πίνακα και να δει πως εφαρμόζεται ο αλγόριθμος για διάφορους μη ταξινομημένους πίνακες, προσφέροντας παράλληλα και μια πολύ καλή οπτικοποίηση του αλγορίθμου. Επίσης θέτονται ερωτήματα-προβλήματα στο υποκείμενο όπως για παράδειγμα *«Μπορείς να σκεφτείς ώστε να δώσεις έναν πίνακα στο applet και να γίνεται αντιμετάθεση σε κάθε σύγκριση;»*. Για την καλύτερη αφομοίωση της γνώσης και για εξάσκηση προτείνεται η λύση ασκήσεων από τους μαθητές στο εκπαιδευτικό λογισμικό «Γλωσσομάθεια».

### **Ζ)Αξιολόγηση**

Προκειμένου να εντοπίσουμε κατά πόσο έγιναν κατανοητά αυτά που παρουσιάστηκαν στο μάθημα, μπορούμε να ζητήσουμε από τους μαθητές να συμπληρώσουν ένα σύντομο test σε μορφή quiz με ερωτήσεις πολλαπλής επιλογής, όπως αυτά που παραθέτουμε μέσα στο λογισμικό. Τα quiz αυτά παρέχουν τη δυνατότητα άμεσης διόρθωσης από το λογισμικό και για κάθε λάθος απάντηση δίνεται ποια ήταν η σωστή με επεξηγήσεις.

## 7.2. Σειριακή Αναζήτηση

**Μάθημα:** Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον

**Τάξη:** Γ΄ Λυκείου

**Γνωστικό Πεδίο:** Σειριακή Αναζήτηση

### Μεθοδολογία διδασκαλίας

Η μεθοδολογία που ακολουθείται στη διδασκαλία της ενότητας βασίζεται στην οπτικοποίηση του αλγόριθμου, την αλληλεπίδραση του μαθητή με αυτόν μέσω του λογισμικού και στις αρχές του εποικοδομητισμού. Ο μαθητής αποτελεί ενεργό μέλος στη διαδικασία της μάθησης αναλαμβάνοντας ο ίδιος πρωτοβουλίες. Σαν πρώτη φάση προκαλείται το ενδιαφέρον των μαθητών μέσα από παραδείγματα της καθημερινότητας, έπειτα ελέγχονται οι πρότερες γνώσεις των εκπαιδευόμενων και γίνεται αναδόμηση των ιδεών τους. Στη συνέχεια μέσα από την αλληλεπίδραση των μαθητών με το λογισμικό «χτίζεται» η καινούρια γνώση. Στο τέλος γίνεται εμπέδωση και εφαρμογή της νέας γνώσης.

### Διδακτικοί Στόχοι

- 1) Οι μαθητές, στα πλαίσια αυτού του μαθήματος, θα πρέπει να καταλάβουν τη χρησιμότητα της αναζήτησης ενός στοιχείου σε έναν πίνακα και τότε την εφαρμόζουμε.
- 2) Να κατανοήσουν ποιοτικά τον τρόπο λειτουργίας του αλγορίθμου της σειριακής αναζήτησης
- 3) Να είναι σε θέση να εφαρμόζουν τον αλγόριθμο αναζήτησης σε ασκήσεις όπου χρειάζεται ταξινόμηση

### Φάσεις διδασκαλίας

#### **A) Έλεγχος προϋπάρχουν γνώσεων**

Πριν περάσουμε στη φάση της διδασκαλίας του αλγορίθμου της αναζήτησης οι μαθητές πρέπει να έχουν κάποιες προηγούμενες γνώσεις. Συγκεκριμένα θα πρέπει:

- 1) Να γνωρίζουν τις μεταβλητές λογικού τύπου και τα είδη των τιμών που μπορεί να πάρει μια τέτοια μεταβλητή.

**Ερώτηση:** Ποιες είναι οι τιμές που μπορεί να πάρει μια λογική μεταβλητή;

- 2) Να έχουν καλή γνώση του τρόπου λειτουργίας της δομής επιλογής **ΑΝ.....ΑΛΛΙΩΣ.....ΤΕΛΟΣ\_ΑΝ** και ευχέρεια στη σύνταξη και τον χειρισμό της

- 3) Να μπορούν να υπολογίσουν το αποτέλεσμα λογικών προτάσεων και να γνωρίζουν τους λογικούς τελεστές.

**Ερώτηση:** Πόσοι και ποιοι είναι οι λογικοί τελεστές και ποια η προτεραιότητα τους;

**Ερώτηση:** Αν  $\alpha =$  αληθής και  $\beta =$  ψευδής να ποια είναι η τιμή του;

$$\lambda \leftarrow \alpha \text{ ΚΑΙ } \beta \text{ Η } \alpha$$

- 4) Να είναι πλήρως εξοικειωμένοι με την έννοια του πίνακα και τον τρόπο προσπέλασης και ανάθεσης τιμών στα κελιά του πίνακα.



**Ερώτηση:**  $A=[4 \ 7 \ 2 \ 1 \ 9 \ 5]$ . Να γίνει αρχικοποίηση και εμφάνιση των στοιχείων του πίνακα  $A$ .

### **B) Αφόρμηση**

Μέσα στο λογισμικό υποθέτουμε ότι ο τηλεφωνικός κατάλογος είναι ένας αρχικοποιημένος πίνακας και θέλουμε να κάνουμε μια αναζήτηση για να βρούμε αν υπάρχει κάπου το όνομα Παπαδόπουλος.

### **Γ) Ανακάλυψη της νέας γνώσης μέσα από το λογισμικό**

Προτού δώσουμε τον αλγόριθμο αναζήτησης σε μορφή ψευδοκώδικα δείχνουμε ποιοτικά πώς λειτουργεί η αναζήτηση σε έναν πίνακα. Υποθέτουμε δηλαδή πάλι ότι αναζητούμε ένα όνομα και ξεκινώντας από το πρώτο στοιχείο του πίνακα ελέγχουμε για το στοιχείο που ψάχνουμε και επαναλαμβάνουμε όσες φορές χρειάζεται μέχρι να βρεθεί το στοιχείο που αναζητούμε. Σε κάθε βήμα και για την καλύτερη αναπαράσταση του αλγορίθμου δείχνουμε και τα αντίστοιχα στιγμιότυπα τρεξίματός του.

### **Δ) Εφαρμογή της νέας γνώσης**

Εφόσον έχουμε εξηγήσει «ποιοτικά» τα βήματα του αλγορίθμου μπορούμε να συνεργαστούμε με τους μαθητές ώστε να βρούμε τις εντολές του αλγορίθμου σε ψευδογλώσσα. Σπάμε τον αλγόριθμο σε μικρότερα προβλήματα όπως για παράδειγμα ποια είναι τα κριτήρια τερματισμού της αναζήτησης, ποια δομή επανάληψης θα χρησιμοποιήσουμε κτλ. Έπειτα δίνεται η δυνατότητα στο μαθητή να λύνει τα επί μέρους προβλήματα καταγράφοντας την απάντηση του σε κατάλληλα πλαίσια που παρέχει το λογισμικό. Έτσι σταδιακά «χτίζεται» η κωδικοποίηση του αλγορίθμου. Στο τέλος αφού ο μαθητής συμπληρώσει τις απαντήσεις του μπορεί να ελέγξει για την ορθότητα τους μέσα από ένα αρχείο με τις λύσεις.

### **Ε) Εμπέδωση της νέας γνώσης**

Μετά από την παρουσίαση της δομής και λειτουργίας του αλγορίθμου αναζήτησης δίνεται η δυνατότητα στους μαθητές να δουν τη λειτουργία της αναζήτησης γραφικά μέσα από ένα Java applet κατασκευασμένο ειδικά για την αναζήτηση. Μέσα από το applet οι μαθητές έχουν την ευκαιρία να αλληλεπιδράσουν με τον αλγόριθμο και απαντώντας σε κάποιες στοχευμένες ερωτήσεις που γίνονται μέσα στο λογισμικό να καταλάβουν πως συμπεριφέρεται ο αλγόριθμος για διάφορες εισόδους. Για την καλύτερη αφομοίωση της γνώσης και για εξάσκηση προτείνεται η λύση ασκήσεων από τους μαθητές στο εκπαιδευτικό λογισμικό «Γλωσσομάθεια».

### **Ζ) Αξιολόγηση**

Προκειμένου να εντοπίσουμε κατά πόσο έγιναν κατανοητά αυτά που παρουσιάστηκαν στο μάθημα, μπορούμε να ζητήσουμε από τους μαθητές να συμπληρώσουν ένα σύντομο test σε μορφή quiz με ερωτήσεις πολλαπλής επιλογής, όπως αυτά που παραθέτουμε μέσα στο λογισμικό. Τα quiz αυτά παρέχουν τη δυνατότητα άμεσης διόρθωσης από το λογισμικό και για κάθε λάθος απάντηση δίνεται ποια ήταν η σωστή με επεξηγήσεις.

### 7.3. Δυαδική Αναζήτηση

**Μάθημα:** Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον

**Τάξη:** Γ΄ Λυκείου

**Γνωστικό Πεδίο:** Δυαδική Αναζήτηση

#### Διδακτικοί Στόχοι

- 1) Στο συγκεκριμένο μάθημα στόχος είναι οι μαθητές να διδαχθούν τον αλγόριθμο της δυαδικής αναζήτησης αλλά μόνο ποιοτικά και να μπορούν να τον εφαρμόσουν σε έναν ταξινομημένο πίνακα. Δηλαδή δεν διδάσκονται την κωδικοποίηση του.
- 2) Επίσης πρέπει να καταλάβουν τη διαφορά της δυαδικής αναζήτησης με τη σειριακή και τότε χρησιμοποιούμε την κάθε μια.
- 3) Να είναι σε θέση να εφαρμόζουν τα βήματα του αλγορίθμου σε έναν ταξινομημένο πίνακα.

#### Μεθοδολογία διδασκαλίας

Η μεθοδολογία που ακολουθείται στη διδασκαλία της ενότητας βασίζεται στην οπτικοποίηση του αλγόριθμου, την αλληλεπίδραση του μαθητή με αυτόν μέσω του λογισμικού και στις αρχές του εποικοδομητισμού. Ο μαθητής αποτελεί ενεργό μέλος στη διαδικασία της μάθησης αναλαμβάνοντας ο ίδιος πρωτοβουλίες. Σαν πρώτη φάση προκαλείται το ενδιαφέρον των μαθητών μέσα από παραδείγματα της καθημερινότητας, έπειτα ελέγχονται οι πρότερες γνώσεις των εκπαιδευόμενων και γίνεται αναδόμηση των ιδεών τους. Στη συνέχεια μέσα από την αλληλεπίδραση των μαθητών με το λογισμικό «χτίζεται» η καινούρια γνώση. Στο τέλος γίνεται εμπέδωση και εφαρμογή της νέας γνώσης.

#### Φάσεις διδασκαλίας

##### **A) Έλεγχος προϋπάρχουν γνώσεων**

Πριν από τη διδασκαλία του αλγορίθμου της δυαδικής αναζήτησης πρέπει να βεβαιωθούμε ότι:

- 1) Οι μαθητές έχουν καταλάβει τον αλγόριθμο της σειριακής αναζήτησης για να μπορέσουν να κάνουν την αντιπαράθεση του με αυτόν της δυαδικής αναζήτησης

**Ερώτηση:** Να γίνει η αναζήτηση του 4 στον παρακάτω πίνακα χρησιμοποιώντας την σειριακή αναζήτηση.  $A=[1\ 3\ 5\ 6\ 6\ 4\ 6\ 7\ 7]$

- 2) Να γνωρίζουν τι είναι ένας ταξινομημένος πίνακας και πως ταξινομείται με τον αλγόριθμο της φυσαλίδας

**Ερώτηση:** Να ταξινομηθεί ο παρακάτω πίνακας με τον αλγόριθμο της φυσαλίδας  $A=[3\ 4\ 5\ 1\ 4\ 7\ 8\ 9\ 10]$

##### **B) Αφόρμηση**

Ο αλγόριθμος της δυαδικής αναζήτησης λειτουργεί μόνο πάνω σε ταξινομημένο πίνακα. Ο τρόπος λειτουργίας της δυαδικής αναζήτησης είναι ακριβώς ο τρόπος που ψάχνουμε ένα όνομα σε έναν τηλεφωνικό κατάλογο που είναι ταξινομημένος κατά

αλφαβητική σειρά. Έτσι λοιπόν μέσα από το παράδειγμα αναζήτησης στον τηλεφωνικό κατάλογο γίνεται μια πρώτη παρουσίαση του αλγορίθμου και οι μαθητές μπορούν να πάρουν μια γεύση για τη δυαδική αναζήτηση.

### **Γ) Ανακάλυψη της νέας γνώσης μέσα από το λογισμικό**

Σε αυτό το σημείο το ζητούμενο είναι οι μαθητές να μπορέσουν μέσα από την εμπειρία τους να καταλάβουν τον τρόπο λειτουργίας της δυαδικής αναζήτησης. Για το σκοπό αυτό δίνεται ένας ταξινομημένος πίνακας και ζητείται από τους μαθητές να κάνουν αναζήτηση ενός συγκεκριμένου ονόματος βλέποντας παράλληλα τα στιγμιότυπα τρεξίματος του αλγορίθμου πάνω στο συγκεκριμένο πίνακα.

### **Δ) Εφαρμογή και εμπέδωση της νέας γνώσης**

Εφόσον οι μαθητές έχουν καταλάβει και μπορούν να τρέξουν τον αλγόριθμο για έναν ταξινομημένο πίνακα, τους υποβάλλονται ερωτήματα όπως για παράδειγμα: πόσες επαναλήψεις απαιτούνται για την εύρεση ενός στοιχείου με τον αλγόριθμο της δυαδικής αναζήτησης; Σε ποιες περιπτώσεις χρησιμοποιείται ο αλγόριθμος της σειριακής και σε ποιες ο αλγόριθμος της δυαδικής αναζήτησης; Ποιος είναι πιο γρήγορος σε έναν ταξινομημένο πίνακα; Οι μαθητές έχουν τη δυνατότητα να απαντήσουν σε προκαθορισμένα πλαίσια και έπειτα να δουν τις σωστές απαντήσεις και ενδεχομένως να αναθεωρήσουν κάποιες λανθασμένες γνώσεις. Για την καλύτερη κατανόηση και εμπέδωση του τρόπου λειτουργίας οι μαθητές μπορούν να αλληλεπιδράσουν με συγκεκριμένο applet για την δυαδική αναζήτηση και μέσα από αυτήν την αλληλεπίδραση να δουν τον αλγόριθμο να εφαρμόζεται σε πραγματικό χρόνο και σε τυχαίους πίνακες.

## 8. Βιβλιογραφία

**Κόκκοτας Π.** (1996), Διδακτικές στρατηγικές για εννοιολογικές αλλαγές στις Φυσικές *Επιστήμες*. Στο Ματσαγγούρας Η. (επιμ.), *Η εξέλιξη της Διδακτικής. Επιστημολογική Θεώρηση*. Αθήνα: Gutenberg.

**Κόκκοτας Π.** (1998), Σύγχρονες προσεγγίσεις στη διδασκαλία των Φυσικών Επιστημών. Η εποικοδομητική προσέγγιση της διδασκαλίας και της μάθησης, Αθήνα: Gutenberg.

**Κόμης Β.** (2001), Διδακτική της Πληροφορικής, Εκδόσεις Πανεπιστημίου Πατρών, Πάτρα.

**Μακράκης Β., Στεφάνου Χ.,** Αναπτύσσοντας αλληλεπιδραστικά μαθησιακά περιβάλλοντα στην ανοικτή και εξ αποστάσεως εκπαίδευση με την τεχνολογία Java και Java Applets.

**Μάργαρης, Α. Παπαστεργίου Μ.,** Εισάγοντας αρχάριους στον προγραμματισμό με τα περιβάλλοντα Kara: Μια προσέγγιση βασισμένη στη θεωρία υπολογισμού.

**Παιδαγωγικό Ινστιτούτο,** Αναλυτικό Πρόγραμμα Σπουδών Πληροφορικής.

**Πανέτσος Σ, Κατσίρης Ι.,** Η Διδασκαλία με Χρήση ΤΠΕ Βασιζόμενη στην Εποικοδομητική Προσέγγιση Η Περίπτωση του Ακολουθιακού Ψηφιακού Κυκλώματος.

**Πανσεληνάς Γ., Κόμης Β.,** Πρωτοβουλίες χρήσης εκπαιδευτικού λογισμικού από τους μαθητές: ένα στοιχείο αυτορρύθμισης κατά τη μάθηση με υποστήριξη.

**Πέτρου Α., Δημητρακοπούλου Α.** Διδασκαλία Πληροφορικής σε μαθητές με κινησιακές αναπηρίες με τη χρήση συνεργατικών σεναρίων σε Διαδικτυακή πλατφόρμα.

**Σιώζου Σ., Τσέλιος Ν., Κόμης Β.,** Επίδραση της χρήσης πολλαπλών αναπαραστάσεων σε εκπαιδευτικό περιβάλλον προγραμματισμού.

**Τζιμογιάννης Α,** Η διδασκαλία του προγραμματισμού στο ενιαίο λύκειο: προς ένα ολοκληρωμένο πλαίσιο με στόχο την ανάπτυξη δεξιοτήτων επίλυσης προβλημάτων.

**Χρυσafiάδη Κ., Βίρβου Μ.,** Η εξ' αποστάσεως διδασκαλία του προγραμματισμού μέσω ενός προσωποποιημένου στις ανάγκες του μαθητή περιβάλλοντος.